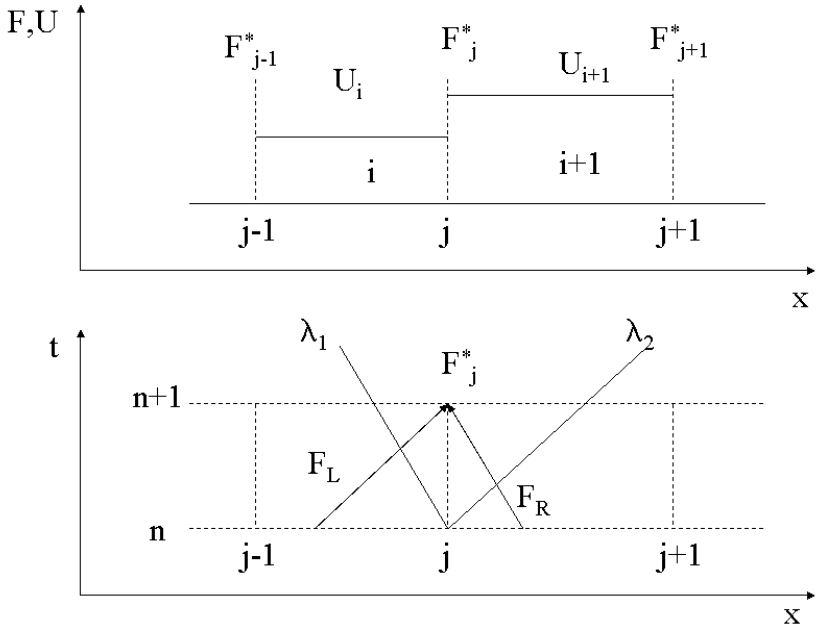
JRC Scientific and Technical Reports

Modelling Shallow Water Flows by a High Resolution Riemann Solver

by
Giovanni Franchello



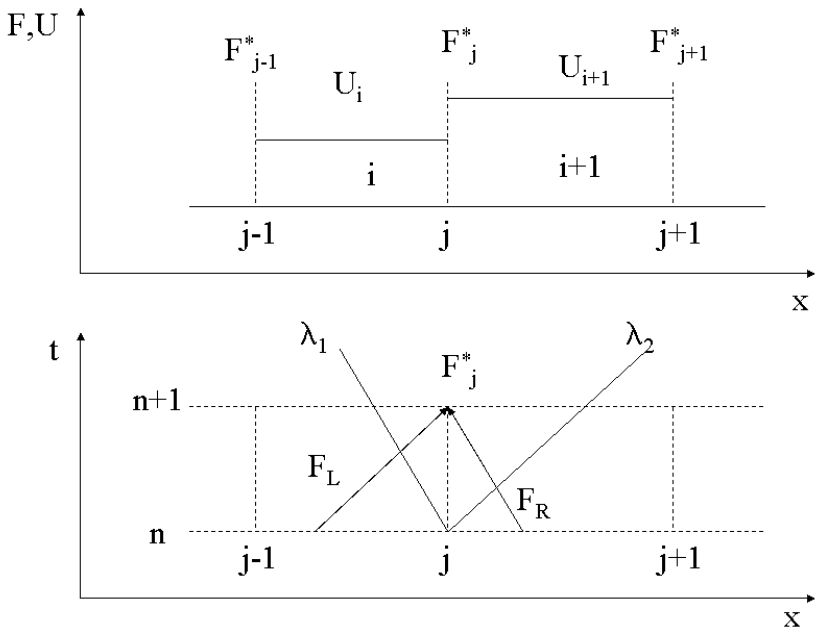
EUR 23307 EN - 2008

JRC EUROPEAN COMMISSION

ipsc Institute for the Protection and Security of the Citizen

Modelling Shallow Water Flows by a High Resolution Riemann Solver

by
Giovanni Franchello



EUR 23307 EN - 2008

JRC
EUROPEAN COMMISSION

ipsc
Institute for the Protection and Security of the Citizen

The Institute for the Protection and Security of the Citizen provides research-based, systems-oriented support to EU policies so as to protect the citizen against economic and technological risk. The Institute maintains and develops its expertise and networks in information, communication, space and engineering technologies in support of its mission. The strong cross-fertilisation between its nuclear and non-nuclear activities strengthens the expertise it can bring to the benefit of customers in both domains.

European Commission
Joint Research Centre
Institute for the Protection and Security of the Citizen

Contact information
Address: Via E. Fermi 2749, I-21027, Ispra, Italy
E-mail: giovanni.franchello@jrc.it
Tel.: +39.0332.785066
Fax: +39.0332.789007

http://ipsc.jrc.ec.europa.eu/
http://www.jrc.ec.europa.eu/

JRC 35059
EUR 23307 EN
ISSN 1018-5593

Luxembourg: Office for Official Publications of the European Communities

Printed in Italy

1 – Introduction

Finite difference methods used for many years for solving the shallow water equations do not conserve mass and require special front-tracking schemes to resolve discontinuities. Finite elements, which only conserve mass globally and not locally, show spurious oscillations at flow discontinuities unless artificial dissipation are employed. Finite volume methods are based on the integral form of the governing equations, which are solved in computational cells. Therefore, mass and momentum are conserved locally, even in the presence of shock waves and other flow discontinuities. A good discussion on this subject can be found in Toro [8], LeVeque [14] and Scott F. Bradford [15]

The "Riemann solvers", or Godunov-type methods, resolve piecewise constant initial Riemann problems – which are, for the shallow water equations, dam-break problems - into a set of propagating waves. This approach is possible for any hyperbolic system of conservation equations: the eigenvalues (characteristic velocity) and the corresponding eigenvectors of the coefficient matrix of the governing equations can be expressed analytically as algebraic functions of the major dependent flow parameters, allowing the application of a technique which makes explicit use of the characteristic directions of the flow fields. Fluxes are evaluated at cell interfaces by solving a Riemann problem, which accurately captures wave propagation and has proven to be efficient and robust. This approach is useful for problems with discontinuous as well as for smooth solutions [7,8].

The experience gained in modelling two-phase flows by Ruel, Städtke et al. [1,2,3,4,5] is now exploited into a more simple case, as far as the shallow water equations involve only 2 equations. The proposed model for solving the shallow water equations is a finite volume Riemann Solver, together with a Flux Vector Splitting technique and a second order scheme. The numerical scheme is explicit, but with implicit treatment of the source terms. Entropy fix in case of transonic flows is also included. Proper models to handle the boundary conditions as well as steps in the bottom are included. The model is able to handle wet as well as dry bed states, including a simple technique for the shoreline tracking, resulting in low numerical diffusion and low numerical viscosity.

The model has been validated with respect to different numerical test cases. Comparisons with the exact solution of the Riemann problem are presented. The paper describes the 1 D numerical scheme and the validation with 1 D test cases. This is done in order to focus directly to the physics of the problems. Furthermore because the concept developed for the 1D model is the same of those for 2D. The model has been implemented in a 2 D system with real topography and bathymetry, allowing simulation of the hydrodynamics with high accuracy. The 2 D model, together with test cases on real Dam break problems, tsunami propagation and long waver run-up problems will be presented in a future paper.

2 – The Shallow Water Equation

The movement of an incompressible fluid with constant density under the influence of a gravitational body force is considered. The description is basically inviscid except for the possible inclusion of a viscous bottom friction term. Vertical accelerations of the fluid are neglected, which allows the integration of the remaining part of the vertical momentum equation, obtaining an expression for the pressure which in turn can then be eliminated from the system. Physically, the horizontal velocity that is retained can be interpreted as a vertical average of the fluid velocity. Recommended reading on the shallow water equation can be found in the book of Toro [8].

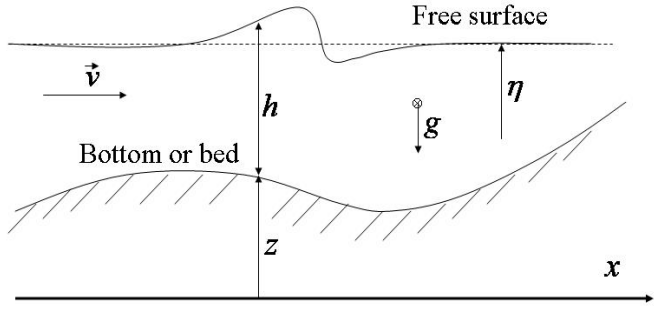


Fig. 2.1 - Schematic of the coordinates and variables of the shallow water model

The 1 D system of equations that governs the above class of flows can be conveniently written in a matrix notation as follows:

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} = C \qquad (2.1)$$

where U is the conservative variables vector, F the flux vector and C the source vector.

$$U = \begin{Bmatrix} h \\ hv \end{Bmatrix}; \qquad F = \begin{Bmatrix} hv \\ hv^2 + gh^2/2 \end{Bmatrix}; \qquad C = \begin{Bmatrix} 0 \\ -gh\left(\frac{\partial z}{\partial x} + s_f\right) \end{Bmatrix} \tag{2.2}$$

In the above notation, h signifies the water depth, v the horizontal velocity of the fluid, z the vertical coordinate of the bottom (or bed), η the elevation of the free surface, g the gravitational acceleration (opposite to the z direction) and S_f denotes the bottom friction that can be expressed by the well known Manning formula

$$s_f = \frac{n^2 v|v|}{h^{4/3}} \tag{2.3}$$

where n is an empirical roughness coefficient for the water – called also Mannig factor- which is in the order of 0.01÷0.1, depending on the surface roughness.

3 – Characteristic Analysis of the Governing Equation

In the proposed model, the Flux Vector Splitting technique is applied. Therefore the analysis of the governing equations is done by looking at the coefficient matrix of the flux vector. The characteristic analysis is done on the *homogeneous system*, i.e., without source term

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} = 0 \tag{3.1}$$

Multiplying the left side of each term of the equation (3.1) by the Jacobian $\partial F / \partial U$, the system can be rewritten in the following form

$$\frac{\partial F}{\partial U} \cdot \frac{\partial U}{\partial t} + \frac{\partial F}{\partial U} \cdot \frac{\partial F}{\partial x} = 0 \tag{3.2}$$

By the chain rule, $\frac{\partial F}{\partial U} \cdot \frac{\partial U}{\partial t} = \frac{\partial F}{\partial t}$, the equation (3.2) becomes

$$\frac{\partial F}{\partial t} + G \cdot \frac{\partial F}{\partial x} = 0 \tag{3.3}$$

where the matrix G is evaluated as follows:

$$G = \frac{\partial F}{\partial U} = \frac{\partial F}{\partial W} \cdot \frac{\partial W}{\partial U} = \frac{\partial F}{\partial W} \cdot \left(\frac{\partial U}{\partial W}\right)^{-1} = \begin{Bmatrix} 0 & 1 \\ gh - v^2 & 2v \end{Bmatrix} \tag{3.4}$$

and $W = \{h, v\}$ is the primitives vector. Note that following the above described procedure it is possible to obtain the coefficient matrix of any vector of independent variables, i.e., primitive vector $W = \{h, v\}$; conservative vector $U = \{h, hv\}$; or any set of independent quantities like $V = \{v, c\}$. Such a 'similarity' transformation does not change the eigenvalues of the system of the governing equations.

3.1 – Eigenvalues

The root of the characteristic polynomial

$$|G - \lambda I| = 0 \tag{3.5}$$

lead to the following Λ vector of real and distinct eigenvalues, or characteristic velocity, such that we are dealing with a ***strictly hyperbolic*** system.

$$\Lambda = \begin{Bmatrix} \lambda_1 \\ \lambda_2 \end{Bmatrix} = \begin{Bmatrix} v - \sqrt{gh} \\ v + \sqrt{gh} \end{Bmatrix} = \begin{Bmatrix} v - c \\ v + c \end{Bmatrix} \tag{3.6}$$

where $c=\sqrt{gh}$ is the celerity of the gravitational wave, which takes the place of the speed of sound as in gas dynamics and the Froude number $Fr=|v|/c$ is the analogue to the Mach number. The notation in equation (3.5) is that |X| denotes the *determinant* of the X matrix, I is the identity matrix and λ the unknown quantity of the polynomial.

3.2 – Eigenvectors

Due to his hyperbolicity, the matrix ***G*** can be reduce to a diagonal form, i.e, $G=T\Lambda T^{-1}$. To do this we need the right eigenvectors T_i , which satisfy:

$$\left|G-\lambda_k I\right|\cdot T_k=0 \qquad for \quad k=1,2 \tag{3.7}$$

The eigenvector associated to $\lambda_1=v-c$ is $T_1=\{1/(v-c),1\}$ and with $\lambda_2=v+c$ is $T_1=\{1/(v+c),1\}$
The vectors T_1,T_2 form the columns of the following matrix T

$$T=\begin{Bmatrix}\dfrac{1}{v-c} & \dfrac{1}{v+c}\\ 1 & 1\end{Bmatrix} \tag{3.8}$$

where the inverse is

$$T^{-1}=\frac{1}{2c}\begin{Bmatrix}-c^2+v^2 & c-v\\ c^2-v^2 & c+v\end{Bmatrix} \tag{3.9}$$

3.3 – Split matrix coefficient

Starting from the diagonal form, the matrix G can be split into elementary parts with respect to the individual characteristic velocity

$$\begin{aligned}G=T\cdot\Lambda\cdot T^{-1}&=T\cdot\begin{Bmatrix}\lambda_1 & 0\\ 0 & \lambda_2\end{Bmatrix}\cdot T^{-1}=T\cdot\begin{Bmatrix}\lambda_1 & 0\\ 0 & 0\end{Bmatrix}\cdot T^{-1}+T\cdot\begin{Bmatrix}0 & 0\\ 0 & \lambda_2\end{Bmatrix}\cdot T^{-1}\\ &=\lambda_1\left[T\cdot\begin{Bmatrix}1 & 0\\ 0 & 0\end{Bmatrix}\cdot T^{-1}\right]+\lambda_2\left[T\cdot\begin{Bmatrix}0 & 0\\ 0 & 1\end{Bmatrix}\cdot T^{-1}\right]=\lambda_1G_1+\lambda_2G_2\end{aligned}$$

Summarising, the matrix G is given by

$$G=\sum_{k=1}^{2}\lambda_k G_k \tag{3.10}$$

with

$$G_k=T\cdot I_k\cdot T^{-1} \qquad for \quad k=1,2 \tag{3.11}$$

and where I_k is a null matrix with 1 in position k,k .

The splitting of the matrix coefficient allows to distinguish between the individual contributions of the two wave's propagation process to the change in time of the flux vector. By substitution of Eq. (3.10) into Eq. (3.1), and noting that the sum of the G_k matrices led to the identity matrix, i.e.,

$$\sum_{k=1}^{2}G_k=I \tag{3.12}$$

the following equation is obtained

$$\left(G_1+G_2\right)\frac{\partial F}{\partial t}+\lambda_1G_1\cdot\frac{\partial F}{\partial x}+\lambda_2G_2\cdot\frac{\partial F}{\partial x}=0 \tag{3.13}$$

4 – Numerical Method

For the numerical solution scheme, the governing Eq. (2.1) is transformed into a finite volume and time approximation, as described in Fig. 4.1. Indices *i* indicate the control volumes or cells, *j* the cell interfaces, *n* the time steps level.

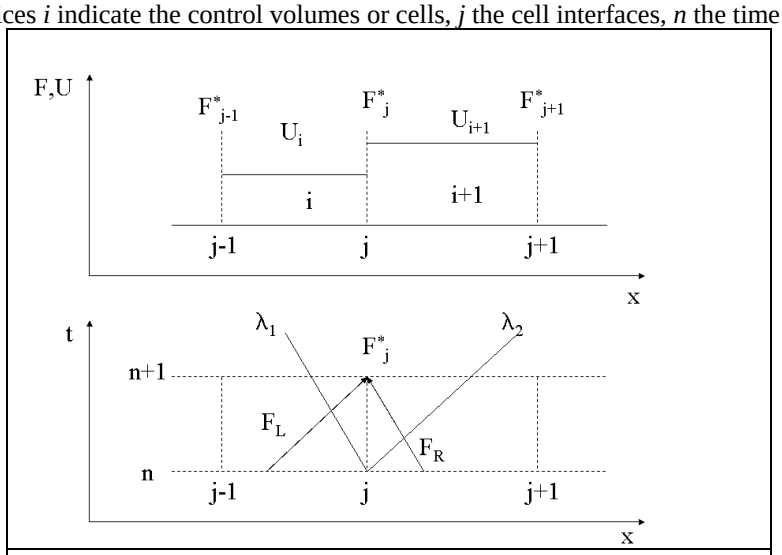


Fig. 4.1: Linearized Riemann Solver for 1D shallow water

The conservative vector, in the cell *i* at time level *n+1*, is given by

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x_i}\left(F_j^{*n} - F_{j-1}^{*n} \right) + C_i^{n+1}\Delta t \qquad (4.1)$$

Note that the source vector is evaluated at the "new" time step $n+1$. The space domain is transformed into so called dam break problems at each cell interface.

4.1 – The solution strategy: the Flux Vector Splitting scheme

By "freezing" the G matrices coefficients in the time step, Eq. (3.12) becomes

$$\sum_{k=1,2}\left(\frac{\partial (G_k F)}{\partial t} + \lambda_k \frac{\partial (G_k F)}{\partial x} \right) = 0 \qquad (4.2)$$

Decoupling we obtain a set of independent ODE $d(G_k F)/dt = 0$ on curves $dx/dt = \lambda_k$ .The solution of the Riemann problem – or dam break problem - for each cell interface is therefore the numerical flux F_j^*, called also "Gudonov " flux, calculated from the corresponding fluxes F_L, F_R in the "left" and "right" cell as

$$F_j^* = \sum_{k=1}^{2} G_k F_K \qquad (4.3)$$

where
$$F_k = \begin{cases} F_L & if\ \lambda_k > 0 \\ F_R & if\ \lambda_k < 0 \end{cases} \quad , \ for \qquad k = 1,2 \qquad (4.4)$$

Looking to the schematisation in Fig 2, where $\lambda_1 < 0$ and $\lambda_2 > 0$, we have $F_1 = F_R$ and $F_2 = F_L$.
Eq. (4.3) and (4.4) can be interpreted as a 'weighting procedure' for the downstream and upstream fluxes based on the sign of the different eingenvalues. Fig. 2 illustrates how the left and right side flux vectors contribute to the weighted-average flux at the cell interface.
This illustrates also the difference from the simpler donor cell technique where the inter-cell state parameters are taken only from the upstream cell based on the sign of the velocity whilst disregarding the signal propagation properties along characteristic lines.

The Eq. (4.4) can be rewritten in the following compact form

$$F_K = \frac{F_L + F_R}{2} + sign(\lambda_k)\frac{F_L - F_R}{2}, \qquad for \quad k = 1,2 \tag{4.5}$$

which in turn can be introduced in Eq. (4.3) obtaining

$$F_j^* = \left(\sum_{k=1}^{2} G_k\right)\frac{F_L + F_R}{2} + \left(\sum_{k=1}^{2} G_k sign(\lambda_k)\right)\frac{F_L - F_R}{2} \tag{4.6}$$

Introducing the G_j^* matrix and replacing G_k matrices by Eq. (3.11), the following is obtained

$$G_j^* = \sum_{k=1}^{2} G_k sign(\lambda_k) = T \cdot \begin{Bmatrix} sign(\lambda_1) & 0 \\ 0 & sign(\lambda_2) \end{Bmatrix} \cdot T^{-1} \tag{4.7}$$

Remembering that $\sum_{k=1}^{2} G_k = I$, Eq. (4.6) can be rewritten in the following compact form

$$F_j^* = \frac{F_L + F_R}{2} + G_j^* \frac{F_L - F_R}{2} \tag{4.8}$$

Replacing in Eq. (4.7) the coefficient of the matrices $T \quad T^{-1}$ and introducing $s_k = sign(\lambda_k)$, the following is obtained

$$G_j^* = \frac{1}{2c}\begin{Bmatrix} (c+v)s_1 + (c-v)s_2 & s_2 - s_1 \\ (c^2 - v^2)(s_2 - s_1) & (c-v)s_1 + (c+v)s_2 \end{Bmatrix} \tag{4.10}$$

The matrix G_j^* is valid for any values of the eigenvalues sign. Particular cases are for "subsonic" flows ($s_1 = -1 \quad s_2 = 1$), and for "supersonic" flow ($s_2 = s_1$), which lead to the following simplified matrices

$$G_j^{sub} = \frac{1}{c}\begin{Bmatrix} -v & 1 \\ c^2 - v^2 & v \end{Bmatrix}, \qquad G_j^{\sup} = s_1 \cdot I \tag{4.11}$$

Note that only in the case of supersonic flow - like in the donor technique - the flux from the upstream cell is taken.

The coefficient of the G_j^* matrix - the velocity *v* and celerity *c* - are computed by an arithmetic average of such quantities at "left" and "right" side of the cell interface. Thus, the method above described is called approximate Riemann solver because the evaluation of the exact physical flux vector is based on an approximate state, i.e.

$$V_j^{\sim} = \begin{Bmatrix} v \\ c \end{Bmatrix} = \frac{V_L + V_R}{2} \tag{4.12}$$

The approximate Riemann solvers differ from the exact Riemann solver for its simplicity – which give robustness to the numerical scheme - and for saving computational time. Different techniques based on approximate Riemann solvers exist. In principal they differ on the way the approximate state and the interface flux are evaluated. The "standard" Gudonov-type methods evaluate the interface cell flux $F_j^* = F(U_j^*)$. The Flux Vector Splitting technique proposed here uses the approximated state vector $V_j^{\sim}$ in order to evaluate the G_j^* matrix. The flux at the interface is therefore **direct function of the G_j^* matrix and the fluxes at "left" and "right" cell, F_L, F_R.** The method explained above can be applied in any hyperbolic system, with disregard to the number of equations involved. It is convenient when we have to deal with big matrices, such in modelling two phase flows. Also for shallow flows the method is attractive for his simplicity and compactness. It can be easily expanded into a 2 Dimensions scheme. Additional equations for the transport of contaminants and/or bed modification can be introduced without drastically increase the complexity of the numerical scheme.

The technique proposed here is similar to the Flux Vector Splitting (FVS) technique used in gas dynamic for the Euler equations because the flux at the cell interface is calculated by splitting the cell fluxes according to the sign of the eigenvalues. The FVS technique requires the homogeneity property (i.e. $F = GU$) which is not satisfied for the shallow water equations. Our technique is an alternative way to calculate the numerical fluxes at the cell boundaries, without to require the homogeneity property as for the original FVS technique [5].

Appling the procedure described for the Flux Vector Splitting (Section 3 and Section 4.1) on the vector $V=\{v,c\}$, the following results are obtained

$$V_j^* = \frac{V_L + V_R}{2} + H_j^* \frac{V_L - V_R}{2} \tag{4.13}$$

with $H_j^* = \begin{Bmatrix} (s_1+s_2)/2 & s_2 - s_1 \\ (s_2-s_1)/4 & (s_1+s_2)/2 \end{Bmatrix}$ (4.14)

and in particular for subsonic and supersonic flow

$$H_j^{sub} = \begin{Bmatrix} 0 & 2 \\ \frac{1}{2} & 0 \end{Bmatrix}, \qquad H_j^{\sup} = s_1 \cdot I \tag{4.15}$$

The results obtained for the subsonic case correspond to the Exact Riemman Solution in case of two-rarefaction waves [8] and will be used to calculate the flux in the case of a near dry bed as well as in the case of a critical flow.

4.2 – The final solution and the treatment of the source Terms

After having evaluated the fluxes at the cell interface, we are ready to calculate the state property vector at the new time step, including the effect of the source terms. The strategy adopted consists on first to evaluate the fluxes based on the *homogeneous pure advection problem, Eq* (3.1)

$$U_i^{(adv)} = U_i^n - \frac{\Delta t}{\Delta x_i}\left(\overset{n}{F}_j^* - \overset{n}{F}_{j-1}^* \right) \tag{4.16}$$

and then add the source term and evaluate the conservative vector at the new time level, including the sources as function of the state property at the new time level.

$$U_i^{n+1} = U_i^{(adv)} + C_i^{n+1}\Delta t \tag{4.17}$$

Since the source term is 0, the quantity $h^{n+1} = h^{(adv)}$.

The velocity at the new time level, in the case of inviscid flow, including only the bed slope, is obtained as follows:

$$v_i^{inv} = \frac{(hv)_i^{adv} - g h_i^{n+1} \frac{\partial z}{\partial x}\Delta t}{h_i^{n+1}} = \frac{(hv)_i^{adv}}{h_i^{n+1}} - g\left(\frac{\partial z}{\partial x}\right)_i \Delta t \tag{4.18}$$

In the case of viscid flow, where the bed roughness is taken into account, the velocity at the new time level is

$$v_i^{n+1} = v_i^{inv} - g\frac{n^2 \left(v_i^{n+1}\right)^2 sign\left(v_i^{n+1}\right)}{\left(h_i^{n+1}\right)^{4/3}}\Delta t \tag{4.19}$$

Because the sign of the velocity at the new time level maintains the sign of the velocity in case of inviscid flow, Eq. (4.19) can be rewritten in the following form

$$\left|v_i^{n+1}\right| = \left|v_i^{inv}\right| - A\left|v_i^{n+1}\right|^2 \tag{4.20}$$

where $A = g\frac{n^2}{\left(h_i^{n+1}\right)^{4/3}}\Delta t$ (4.21)

The unique possible solution of the second order Eq. (4.20) is the following

$$v_i^{n+1} = sign\left(\frac{-1+\sqrt{1+4A\left|v_i^{inv}\right|}}{2A}, v^{inv} \right) \tag{4.22}$$

4.3 – Critical flow

In case of a critical flow or sonic rarefaction, care is taken in evaluating the interface flux. In such a case, where one of the eigenvalues can be zero, the sign of the propagation wave correspondent to such eigenvalue cannot be captured, so the

'weighting' scheme of the left and right flux vectors, on which the flux vector splitting scheme is based cannot be applied. The strategy adopted is simply to assign the sonic conditions at the cell interface and then evaluate the corresponding flux. Starting from the state vector V_j^* calculated by Eq. (4.13), and imposing the conservation of the momentum by fixing the sonic condition, the following equation is obtained

$$v\frac{c^2}{g} = s\frac{s^2}{g} \tag{4.23}$$

where in the LHS of the equation is the momentum at the approximate state, while in the RHS is the momentum at sonic sonic condition, i.e., $v = c = \sqrt{gh} = s$. The sonic speed is so given by

$$s = \left(vc^2\right)^{1/3} \tag{4.24}$$

The state vector at sonic condition becomes

$$V_j^{sonic} = \begin{Bmatrix} sign(v)\cdot s \\ s \end{Bmatrix} \tag{4.25}$$

The flux is then evaluated as direct function of the state vector, i.e, $F_j^* = F\left(V_j^{sonic}\right)$ Such a scheme is applied when a transonic rarefaction condition is seen, i.e. when are satisfied one of the following conditions

a. $Fr_L < 1 \quad and \quad Fr_R > 1 \quad and \quad h_L > h_R$ (4.26)

b. $Fr_R < 1 \quad and \quad Fr_L > 1 \quad and \quad h_R > h_L$ (4.27)

4.4 – Dry bed state

In the case the water depth is close to zero (dry bed) care is taken in order to handle the particular problem. The strategy adopted is 1) set the admissible water depth in the cells to a minimum value $h_{\min} = 10^{-6}$ m and 2) handle the cases when the calculated water depth **at the cell interface** is below an admissible value $h_{crit} = 0.1$. Such conditions can occur in case of two opposite rarefaction waves. The interface flux is calculated as a direct function of the V_j^* vector calculated by Eq. (4.13), i.e., $F_j^* = F\left(V_j^*\right)$. Note that in this case the matrix H_j^* (see Eq. 4.14 and 4.15) contains only pure numbers, i.e., there is not a division by the celerity c like in the matrix G_j^* (see Eq. 4.10 and 4.11). In case a negative celerity is calculated ($c_j^* < 0$) the flux at cell interface is set to zero. In case that in the left or right side of the cell interface the bed is dry, the flow is considered critical: the interface flux is calculated by using the flow condition of the wet cell and applying the scheme described in Section 4.3. This strategy has proven to be very robust and efficient, mainly in case of wetting processes.

4.5 – Second-order scheme

A second order scheme has been implemented by a linear reconstruction of the solution in all computational cells following the Monotonic Upstream Scheme for Conservative Laws (MUSCL) approach, as indicated schematically in Fig. 4.2

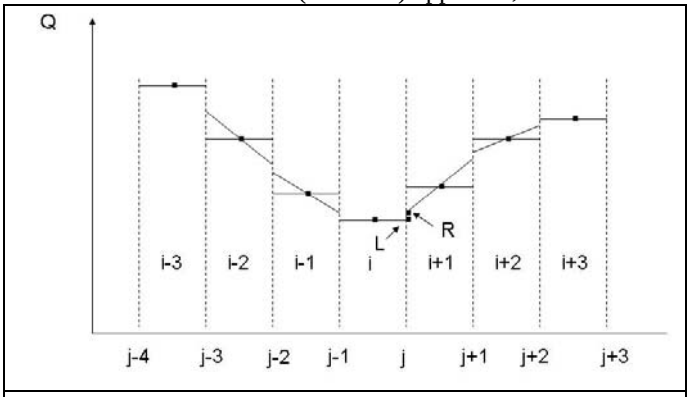


Fig.4.2: Linear reconstruction of the solution

Values for the vector $Q = \{\eta, v, hv\}$ at the left and right side of the cell interfaces j are calculated by a linear extrapolation from the adjacent cells as

$$Q_{jL} = Q_i + \sigma_i \frac{\Delta x_i}{2}$$
$$Q_{jR} = Q_{i+1} - \sigma_{i+1} \frac{\Delta x_{i+1}}{2} \tag{4.26}$$

The quantity η is the elevation of the free surface,. σ_i are the slopes of the quantities Q_i "processed" by a modified 'MINMOD' limiter function in order to maintain a monotonic behaviour of the solution

$$\sigma_i = f_{\lim}[\sigma_{iL}, \sigma_{iR}] = \begin{cases} 0 & if \quad \sigma_{iL}\sigma_{iR} < 0 \\ \sigma_{iL}\sigma_{iR}/(\mathrm{f}\ \sigma_{iL} + \sigma_{iR}) & if \quad |\sigma_{iL}| < |\sigma_{iR}| \\ \sigma_{iL}\sigma_{iR}/(\sigma_{iL} + \mathrm{f}\ \sigma_{iR}) & \text{if} \quad |\sigma_{iL}| > |\sigma_{iR}| \end{cases} \tag{4.27}$$

where $\sigma_{iL} = \left(\frac{\partial Q}{\partial x}\right)_{i-1} = \frac{Q_i - Q_{i-1}}{x_i - x_{i-1}}; \sigma_{iR} = \left(\frac{\partial Q}{\partial x}\right)_{i+1} = \frac{Q_{i+1} - Q_i}{x_{i+1} - x_i}$. The f constant has been introduced in order to avoid noise diffusion and overshooting which can be triggered by the second order scheme. Note that setting *f=0* the original 'MINMOD' limiter function is applied: for $f \to \infty$ the slope $\sigma_i \to 0$. Setting *f=1*, then the limited slope is the inverse of the sum of the slope inverses:

$$\sigma_i = \frac{1}{1/\sigma_{iL} + 1/\sigma_{iR}} \tag{4.28}$$

The primitive vector $W_k = \{h_k, v_k\}$ is then obtained from the extrapolated quantity vector $Q_k = \{\eta_k, v_k, (hv)_k\}$ as follows

$$h_k = \eta_k - z_k$$
$$v_k = \begin{cases} \frac{(hv)_k}{h_k} & if \quad h_k > h_{\lim} \\ v_k & if \quad h_k < h_{\lim} \end{cases} \tag{4.29}$$

where $h_{\lim} = 0.1$ is a limiting value chosen to handle the 'near dry bed'. The quantity z_k is evaluated as follows

$$z_L = z_i + \left(\frac{dz}{dx}\right)_i \frac{\Delta x_i}{2}$$
$$z_R = z_{i+1} - \left(\frac{dz}{dx}\right)_{i+1} \frac{\Delta x_{i+1}}{2} \tag{4.30}$$

The method of handling the surface gradient has been proved to be efficient and robust because at near steady state the source due to the bathymetry are balanced by the flux gradient (Zhou [23]). The method used by LeVeque [24] on balancing source terms and flux gradients consists on incorporating the sources in the fluxes [14,24], providing good simulation of small perturbation propagation. However, for the surface gradient model proposed here, particular cases arise when:

a) the extrapolated $Z_L \neq Z_R$, i.e., a jump of the bottom can occurs in the cell interface.

b) the extrapolated water depth h_k is negative: this happens during a wetting/drying process which can be seen as a moving shoreline, where the shoreline is the interface between dry and wet cells, i.e., it can be defined as the intersection between the bottom and the free surface.

Such singularities are treated separately by special models explained in the following sections.

4.6 – Boundary Conditions

Several boundary conditions model are needed in order to specify the physical problem: in all the cases the numerical flux at the cell interface is calculated by adding a fictitious cell (the called also "ghost" volume) in the side where the cell is missing and assigning a proper primitive state vector W^g. The following situations are considered:

a) Solid reflective boundary. $W_L^{\,g} = \{h_R, -v_R\}$ $W_R^{\,g} = \{h_L, -v_L\}$

b) Fully tranmissive boundary. $W_L^{\,g} = \{h_R, v_R\}$ $W_R^{\,g} = \{h_L, v_L\}$

c) Partially tranmissive boundary (the water depth is fixed). $W_L^{\,g} = \{h_L, v_R\}$ $W_R^{\,g} = \{h_R, v_L\}$

If a fictitious cell is added to the right side, the numerical flux in the boundary is given by

$$F_j^B(W_L) = F_j^*(W_L, W_R^g) \tag{4.31}$$

If a fictitious cell is added to the left side, the numerical flux in the boundary is given by

$$F_j^B(W_R) = F_j^*(W_L^g, W_R) \tag{4.32}$$

4.7 – Bottom step

A straightforward way to evaluate the bottom geometry is from the junction elevations: the bottom elevations and slopes are calculated by linear interpolation:

$$z_i = \frac{z_j + z_{j-1}}{2}$$
$$\left(\frac{\partial z}{\partial x}\right)_i = \frac{z_j - z_{j-1}}{\Delta x_i} \tag{4.33}$$

By such a procedure no steps in the bottom elevation occur at the cell interface, i.e. $z_L = z_R = z_j$. The disadvantages rise for the case of the 2 D model: the interpolated bottom elevation in the x-direction can not be the same than the interpolated bottom elevation in the y-direction.

A more general way to define the bottom geometry - applicable also in 2 D – can be achieved by defining explicitly the elevations z_i and the slopes $\left(dz/dx\right)_i$ at each cell centre, and the elevations z_j at each junctions. Using this methodology, steps can occur at the cell interface, i.e., $z_L \neq z_R \neq z_j$ (see Fig. 4.3). The slopes $\left(dz/dx\right)_i$ can be calculated by Eq. (4.27), setting $f = 0$ (MIMMOD limiter).

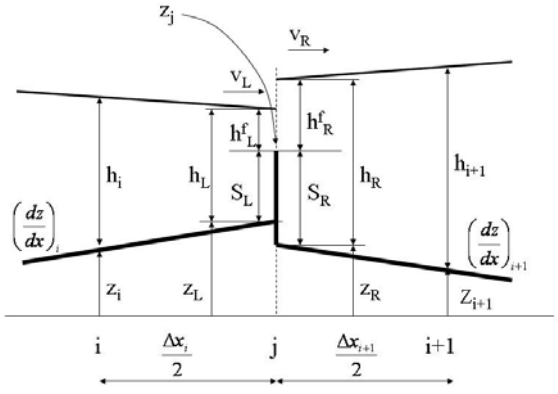


Fig.4.3: Bottom discontinuities at the cell interface

The strategy followed in modelling such steps at the cell interface is to consider the left and right sides as a combination of solid reflective boundary with free flowing flow. In Fig. 4.3 the following quantities are introduced:

$$S_L = \max(z_j - z_L, 0); \quad S_R = \max(z_j - z_R, 0); \quad h_L^f = h_L - S_L; \quad h_R^f = h_R - S_R;$$

The numerical flux for the free flowing layer is given by:

$$\left(F_j^*\right)^f = F_j^*\left(W_L^f, W_R^f\right) \tag{4.34}$$

where $W_L^f = \left\{h_L^f, v_L\right\}; \quad W_R^f = \left\{h_R^f, v_R\right\};$

The numerical flux for the left volume (step at right) is calculated by integrating Eq. (4.31) for the step S_L and adding the flux for the free flowing layer as in Eq. (4.34)

$$\left(F_j^*\right)_L = F_j^B(W_L) - F_j^B(W_L^f) + F_j^*(W_L^f, W_R^f) \tag{4.35}$$

Note that it incorrect to directly apply the Eq. (4.31) to the step S_L because we obtain a numerical flux for a layer with the free surface at S_L which does not correspond to the physical problem. Defining $W_L^S = \{S_L, v_L\}$; we obtain a numerical flux $F_j^B(W_L^S)$ which is different to the difference $F_j^B(W_L) - F_j^B(W_L^f)$, that in turn represents the numerical flux of a layer where the bottom is at h_L from the free surface and the top is at h_L^f from the same free surface. Such differences are not negligible when the step S_L and the difference $h_L^f = h_L - S_L$ are in the same order of magnitude.

The numerical flux for the right volume (step at left) is calculated by integration of Eq. (4.32) for the step S_R and adding the flux for the free flowing layer as in Eq. (4.34)

$$\left(F_j^*\right)_R = F_j^B(W_R) - F_j^B(W_R^f) + F_j^*(W_L^f, W_R^f) \tag{4.36}$$

The same discussion about the application of Eq. (4.31) to the step S_L can be repeated for the application of the Eq. (4.32) to the step S_R .

The implementation of such strategy gives the possibility to simulate complex geometry, like dams, dikes, breakwaters and other submerged barriers, without increasing the number of cells or mesh refinement.

4.8 – Interface between dry and wet cells

A non trivial problem is the modelling of the interface between dry and wet cells. Such a situation can be seen when a front is arriving in a dry cell, or when a wet cell is becoming dry. Such physical problem is called shore line tracking.. In the cell where the shore is situated, we can have at one side a completely wet cell and at the other side a completely dry cell. A very simple way to handle the physical problem is to increase the number of cells at such interface. This approach is costly in terms of computer modelling, CPU time and memory space.

Several studies have been addressed in the shoreline tracking, or preserving depth positivity, by Audusse E [17], Fabien Marche[18] and Pilar Brufau[22]. They reconstruct the bed elevation and the water depth at the cell interface in a way that the positivity of the water depth is satisfied; the source terms are defined in a manner that are balanced by fluxes at near zero velocity. XinJian Chen [19] proposed a model to solve flux-based finite difference equations in the 3 D Cartesian-coordinate system (x,y,z). In the model a bilinear bottom surface is used to fit the bottom topography, so that is kept track the dynamic position of the shoreline.

We have tried to implement the model proposed by Marche [17], Audusse[18] and Brufau [22], but serious problems arises when a second order scheme is used and/or in simulating a run-up process. **The model proposed here consists on preserving depth positivity (mass conservation) at near zero velocity but without the reconstruction of the bottom topography**.

As mentioned in Section 4.5 above, the interface between a dry and wet cell is identified when extrapolating the water depth at the cell interface (see Eq. 4.29). Note that such singularity can happen in case a second order scheme as well as a first order scheme is applied. For simplicity, Fig. 4.4 shows the latter case – first order or stationary flow - where the space derivatives of the free surface elevation are zero. Considering the elevation of the free surface η, we note that $\eta = z + h$ only when the cell is completely wet or dry. In case of shore line, the quantity $\eta \neq z + h$, i.e., η can be lower or higher of ***z.***

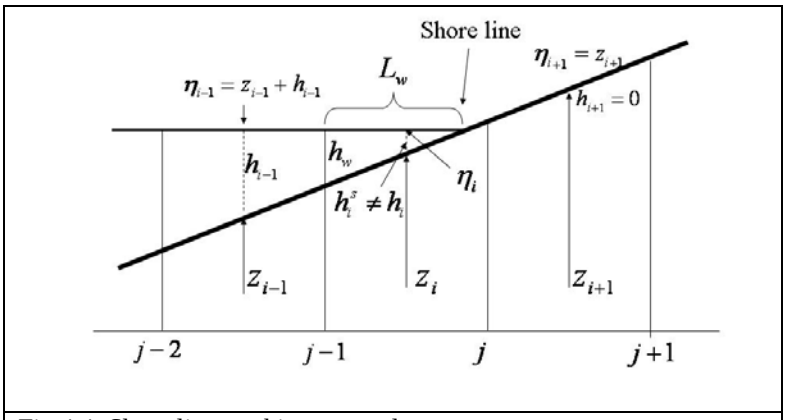


Fig.4.4: Shore line tracking at steady state

The water depth h is intended to be the ratio between the volume of water in the cell and the cell area. In 1 D, the water depth is given by

$$h_i = \frac{L_w h_w}{2\Delta x_i} \tag{4.37}$$

Based on Fig. 4.4, we define the following quantities

$$h_i^{\,s} = \eta_i - z_i; \quad \alpha_i^{\,s} = \left(\frac{\partial \eta}{\partial x}\right)_i - \left(\frac{\partial z}{\partial x}\right)_i \tag{4.38}$$

By simple geometrical considerations, the following expressions can be obtained:

$$\begin{aligned} h_w &= h_i^s + \frac{1}{2}\left|\alpha_i^s\right|\Delta x_i \\ Lw &= \frac{h_w}{\Delta x_i} \end{aligned} \tag{4.39}$$

By replacing Eq. (4.39) in Eq. (4.37) we obtain the following expression for the mean water depth in a cell with a shore line, as function of the free surface elevation and gradient:

$$h_i = \frac{(h_w)^2}{2\left|\alpha_i^s\right|\Delta x_i} = \frac{\left[\eta_i - z_i + \frac{1}{2}\left|\alpha_i^s\right|\Delta x\right]^2}{2\left|\alpha_i^s\right|\Delta x_i} \tag{4.40}$$

Using the absolute value of the quantity $\alpha_i^{\,s}$, Eq (4.38) and (4.39) can be applied in case the cell is wet al left side as well at right side. Note that the quantity h_i^s can be positive as well as negative, while the resulting mean water depth is always positive. With the introduction of the above equations, we demonstrate now that in case of steady state flow $\left(\frac{\partial U_i}{\partial t} = 0\right)$ the flux derivative equal the source terms. The Eq. (4.1) becomes:

$$\frac{1}{\Delta x_i}\left(F_j^* - F_{j-1}^*\right) = C_i \tag{4.41}$$

In the junction j, the flux $F_j^* = 0$, while for the junction $j-1$ we can write the following

$$h_L = h_R = h_w; \quad v_L = v_R = 0; \quad F_L = F_R = F_j^*; \quad \left(\frac{\partial \eta}{\partial x}\right)_i = 0; \quad \left|\alpha_i^s\right| = \left|-\left(\frac{\partial z}{\partial x}\right)_i\right| = \left(\frac{\partial z}{\partial x}\right)_i \tag{4.42}$$

Therefore in the LHS of Eq. (4.41) we have

$$\frac{1}{\Delta x_i}\left(F_j^* - F_{j-1}^*\right) = -\frac{1}{\Delta x_i}\begin{Bmatrix} 0 \\ \frac{1}{2} g (h_w)^2 \end{Bmatrix} = -\frac{1}{2\Delta x_i}\begin{Bmatrix} 0 \\ g (h_w)^2 \end{Bmatrix}$$

while in the RHS

$$C_i = \begin{Bmatrix} 0 \\ -g h_i \left(\frac{\partial z}{\partial x}\right)_i \end{Bmatrix} = \begin{Bmatrix} 0 \\ -g \frac{(h_w)^2}{2\left|\alpha_i^s\right|\Delta x_i}\left(\frac{\partial z}{\partial x}\right)_i \end{Bmatrix} = -\frac{1}{2\Delta x_i}\begin{Bmatrix} 0 \\ g (h_w)^2 \end{Bmatrix}$$

Eq. (4.40) for the mean water depth in a cell with a shore line assure that the flux gradient are balanced by the sources during a steady flow. However, the water depth h, together with the flow velocity v, are independent parameters of the physical problem, calculated at each time step as described in Section 4.2. Moreover, the elevation of the free surface in a cell with the shore line - as well as in the completely wet or dry cell - is needed for the calculation of the slopes as described in Section 4.5.

The quantity η_i can be calculated from Eq. (4.37-39), obtaining:

$$\eta_i = z_i + h_i^{\,s} \tag{4.43}$$

where

$$h_i^s = h_w - \frac{1}{2}\left|\alpha_i^s\right|\Delta x_i \tag{4.44}$$

$$h_w = \sqrt{2\left|\alpha_i^s\right|\Delta x_i h_i} \tag{4.45}$$

Note that in a cell with a shore line, the elevation of the free surface η_i can be lower or higher than the bottom elevation z_i. The quantity h_w is the water depth at the left of the cell if the $\alpha_i^s > 0$, otherwise is the water depth at the right of the cell. The quantity L_w can be used to identify the location of the shore line with high accuracy, i.e., evaluate the percentage of "inundation" of a cell. Such methodology has been proved to be efficient and robust, and it accurately simulates a physical problem without increasing the number of cells.

Eq. (4.43) to (4.45) are also used in case of a moving shoreline.

The methodology explained above for the shoreline tracking consists in finding the intersection between 2 straight lines that identify the bed and the water free surface. In the case of a 2D system the shoreline tracking will consist in finding the intersection between 2 planar surfaces. Substantial differences exist in the Eq. (4.45) for the quantity h_w which requires the solution of a 3rd order polynomial. Due to lack of space in the present paper, shoreline tracking in a 2 D system will be presented in a further publication.

5 – Numerical Tests and comparisons with the Exact Solution

In order to assess the prediction capability of the proposed model, various classical numerical test cases have been chosen and simulation results of the Flux Vector Splitting (FVS) solver are compared with the exact solutions. The selected test cases are summarised in Table 5.1. Part of them (Test 1, 2, 2.1, 3) are "Dam Break Problems" (see Figure 5.1) proposed by Toro [8] and compared with the exact solution provided by the Fortran Program **HL-SWRPEX** available on the web site **http://www.numeritek.com/**. , which is part of the NUMERIKA library [10].

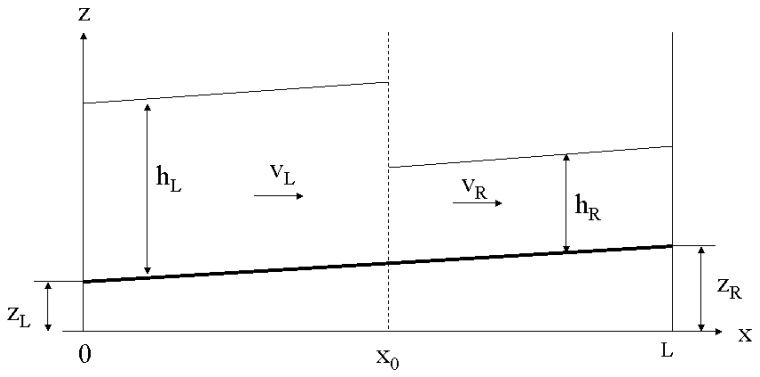


Figure 5.1 – Sketch for tests 1 to 4

Test 4 is a Dam Break problem with **constant bed slope**, proposed by N.Adrianov [13], for which an exact solution can be derived; Test 5 is a Dam Break problem with **bed step**, proposed by Jin and Wen [12] in order to verify the suitability of their method to describe source terms concentrations. Test 1.1, 3.1 and 6 are proposed by the author:
Test 1.1 is a Dam Break problem with water depth and channel length in the order of magnitude of Tsunami wave propagation, Test case 3.1 is derived from Test 3, but with dimensions in the order of magnitude of a real Dam Break: the initial water level at the left is 100 m, while the length of the channel is 50 km. Test 6 is a steady state problem, where the bed elevation has a maxima in the middle of the channel, like a nozzle problem in gas dynamic. Test 7 is a shore line tracking problem proposed in "The third international workshop on long-wave runup models" [20]. The acceleration due to gravity is $g = 9.806 \quad m/s^2$ in all the test cases, except of Test 4 where $g = 2 \quad m/s^2$. The water depth and velocity in table 5.1 are the initial conditions. The water depth is fixed constant during the transient, while the inlet and outlet velocities are updated at each time step by setting up the value of the neighbouring cell, i.e., $v_L = v_1; \quad v_R = v_{Mcells}$. Only for test cases 4 and 6 the inlet and outlet velocities change during the test problem.

Test	h_L [m]	v_L [m/s]	h_R [m]	v_R [m/s]	z_L [m]	z_R [m]	L [m]	x_0 [m]	Mcells	t_{out} [s]
1	1	2.5	0.1	0	0	0	50	10	100	7
1.1	4005	0	4000	0	0	0	$1{\cdot}000*10^3$	$50*10^3$	1000	1800, 3600
2	1	-5	1	5	0	0	50	25	100	2.5
2.1	0.1	-3	0.1	3	0	0	50	25	100	5
3	1	0	10^{-6}	0	0	0	50	20	100	4
3.1	100	0	10^{-6}	0	0	0	$50*10^3$	$20*10^3$	100	400
4	1.3	-2	0.1	-2	0.4	0	1	0.5	300	0.1
5	4	-10	1	-6	0	1	20	10	100	0.5
6	1	0	0.8	0	0	0	50	25	100	50
6.1	1	-	0.5	-	0	0	50	25	100	50
7	Benchmark 1 in "The third international workshop on long-wave ..." http://www.cee.cornell.edu/longwave/									

Table 5.1 – Numerical test cases - The length of the channel (*L*), the position of the initial discontinuity (x_0), the water depth (*h)* and the bed elevation (*z*) are in meters, the velocity (*v*) in m/s, the output time (t_{out}) in sec. Msells is the no. of cells for the numerical simulation.

5.1 – Left rarefactions and right shock

Test case 1 is a classical dam break problem producing a strong right propagation shock wave and a critical left rarefaction wave. Figure 5.1.a shows the comparisons with the exact solution in case that the second order accuracy is NOT applied. Note the small jump at x_0=10m, which correspond to the location of the critical conditions. The resolution of the shock wave (right side) is not too unsatisfactory for a first-order method.

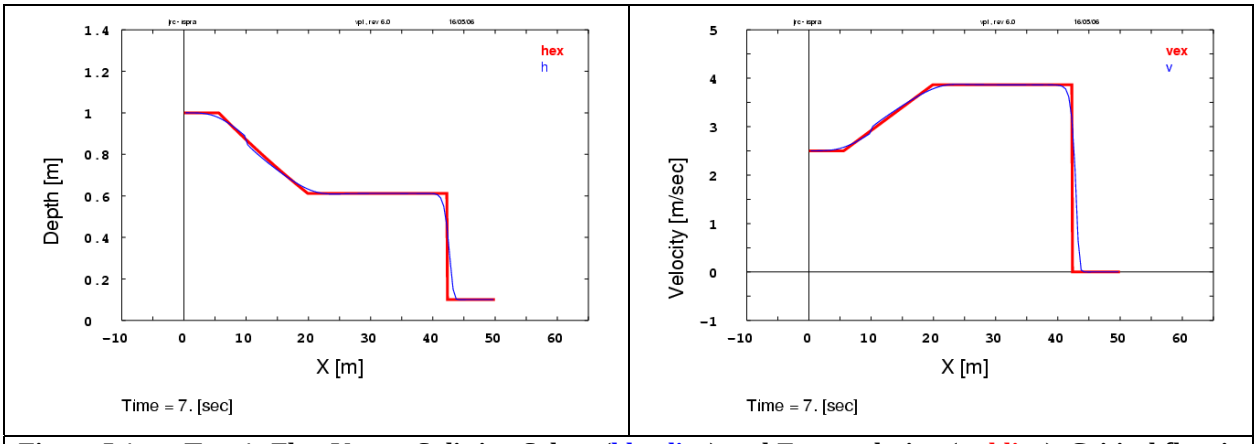


Figure 5.1.a – Test 1: Flux Vector Splitting Solver (blue line) and Exact solution (red line). Critical flow is fixed.

Figure 5.1.b shows results when the second order scheme and MINMOD limiter is applied: the jump at x_0=10m completely disappears, improving the prediction of the sonic rarefaction wave as well as the shock wave. The second order scheme improves fixing the critical flow condition.

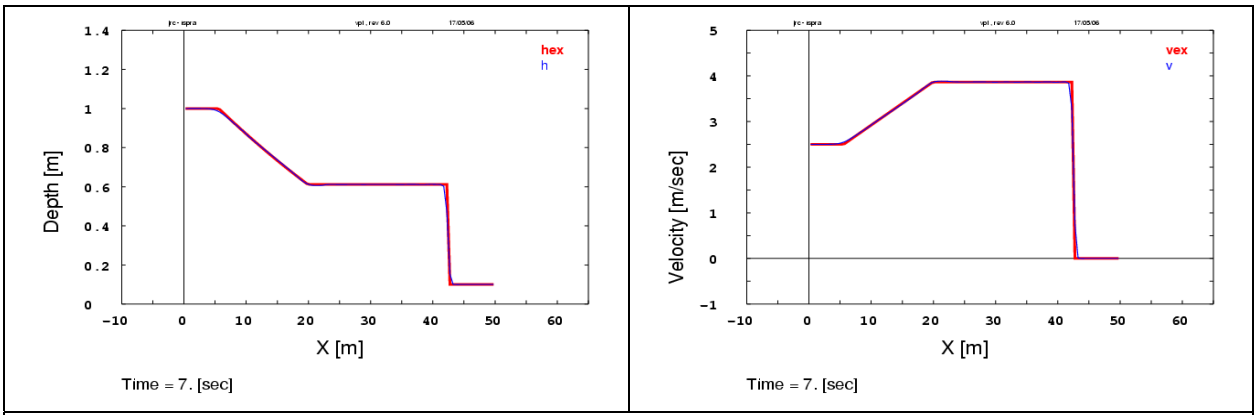


Figure 5.1.b – Test 1: Flux Vector Splitting Solver (blue line) and Exact solution (red line). Second order accuracy with the MINMOD limiter function is applied (f=0).

Test case 1.1 is a 'Dam Break problem' with water depth and channel length in the order of magnitude of Tsunami wave propagation (see Table 5.1). Figure 5.1.d shows the comparison of the simulation with the exact solution for output time 1800 and 3600 sec. The height of the wave is 4002.5 m, against the initial height of 4005 m. The *x* position and height are well simulated, but with a smoothed profile at the wave front, which is increasing with respect to time and x position. Applying the second order scheme (Figure 5.1.e) with the MINMOD limiter function, the wave front is well simulated.

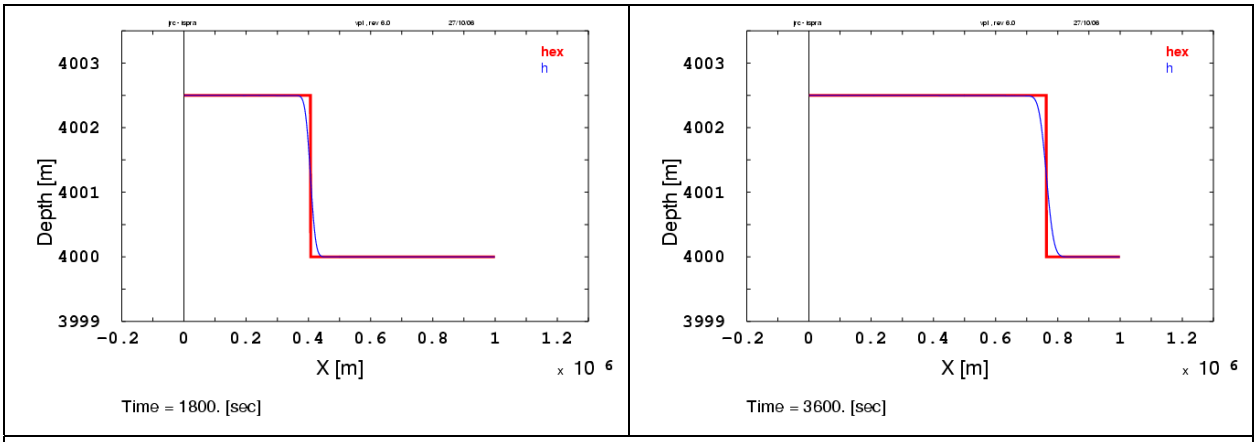


Figure 5.1.d – Test 1.1: Flux Vector Splitting Solver (blue line) and Exact solution (red line).

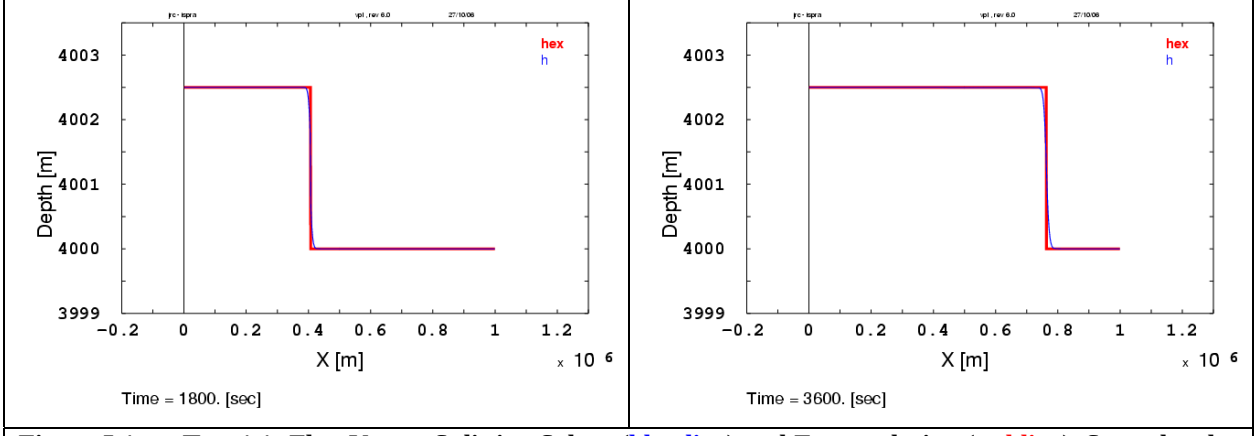


Figure 5.1.e – Test 1.1: Flux Vector Splitting Solver (blue line) and Exact solution (red line). Second order accuracy with MINMOD limiter function is applied (f=0)

5.2 – Two rarefactions waves and dry bed formation

Test cases 2 and 2.1 are chosen with initial condition such that the solution consists of two strong (critical) rarefaction waves producing a near dry bed state (Test 2) and a full dry bed state (Test 2.1). A second order accuracy scheme, together with the MINMOD limiter function is adopted. The minimum admissible water depth in the cells is set to $h_{\min} = 10^{-6} m$. Figure 5.2.a shows the comparisons with the exact solution for the Test 2: small differences are seen only in the near dry state, mainly for the velocity, while for the rest of the wave, the water depth and velocity are well simulated. However, small overshoots at the beginning of the rarefaction waves ($x \cong 5; x \cong 45$) are noted: applying the 'modified' MINMOD limiter function (see Section 4.5) with factor $f = 0.5$, such overshoots disappear (Fig. 5.2.b).

Figure 5.2.c shows results for the Test 2.1, with large dry bed formation. Differences are seen for the velocity in the 'near dry bed' region, while for the rest of the wave the results are still satisfactory, considering also that most of the numerical methods are bound to experience serious difficulties in resolving the flow (Toro, [8])

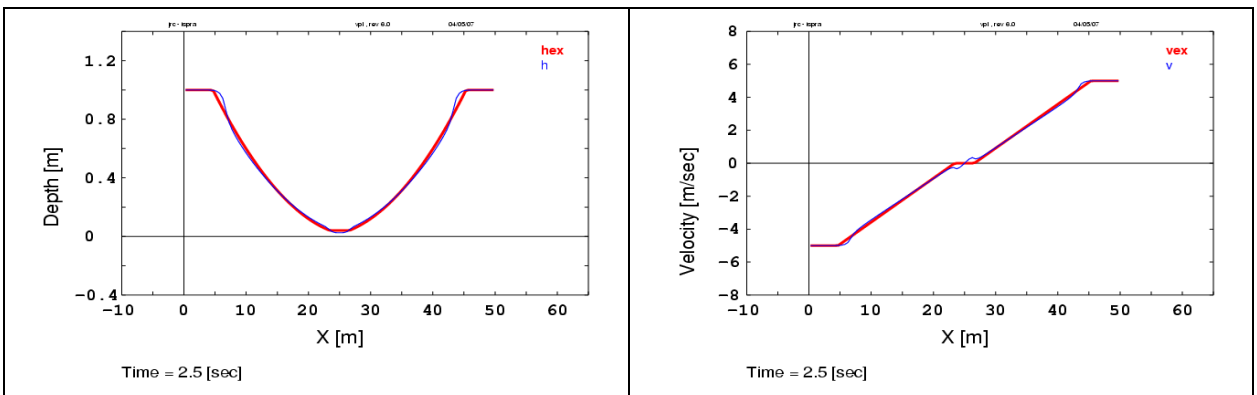


Figure 5.2.a – Test 2: Flux Vector Splitting Solver (blue line) and Exact solution (red line). Second order accuracy with MINMOD limiter function is applied. (f=0)

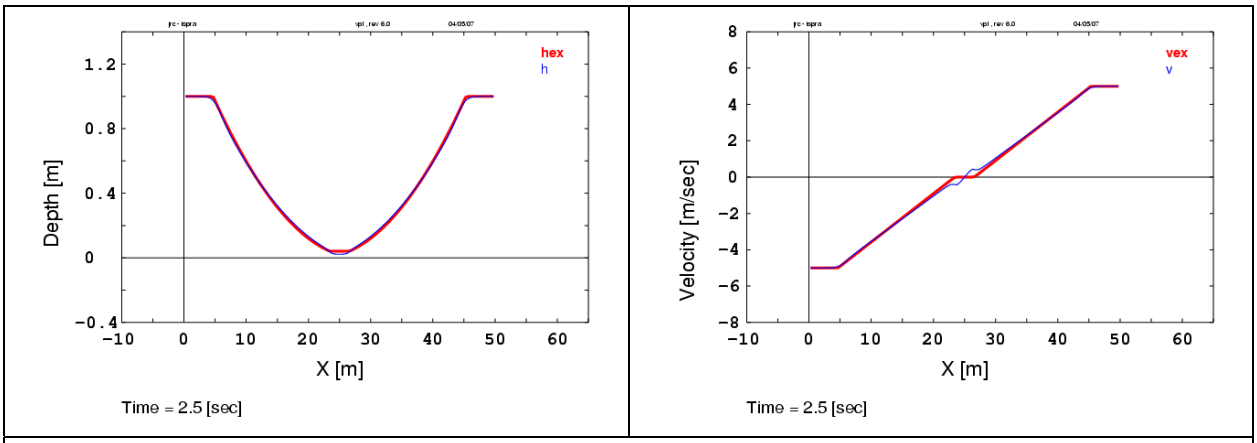


Figure 5.2.b – Test 2: Flux Vector Splitting Solver (blue line) and Exact solution (red line). Second order accuracy with MODIFIED MINMOD limiter function is applied. (f=0.5)

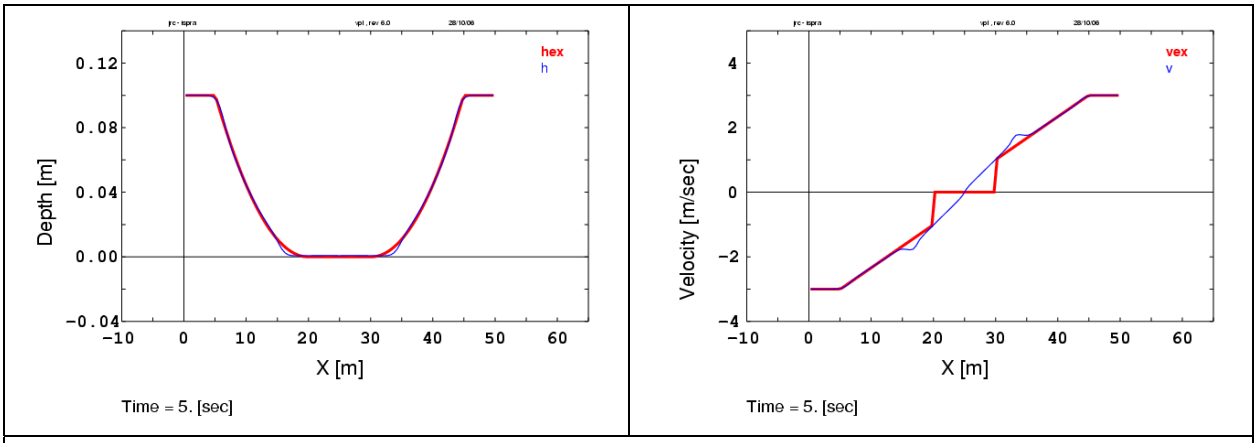


Figure 5.2.c – Test 2.0: Flux Vector Splitting Solver (blue line) and Exact solution (red line). Second order accuracy with MODIFIED MINMOD limiter function is applied (f=0.5)

5.3 – Left critical rarefactions and right dry bed

Test case 3 is a classical "Dam Break Problem", similar to Test case 1, but with a right dry bed state ($h_R = h_{\min} = 10^{-6} m$). Figure 5.3.a shows the comparison between the exact solution and the prediction. A second order scheme with the MINMOD limiter function is applied. There is good agreement for the water depth, while the velocity diverges from the exact solution in the near dry state. Applying the 'modified' MINMOD limiter function (see Section 4.5) with factor $f = 0.5$, the velocity profile is more close to the exact solution (Figure 5.3.b).

Test case 3.1 is derived from Test 3, but with dimensions in the order of magnitude of a real Dam Break: the initial water level at the left is 100 m, while the length of the channel is 50 km. Due to the fact that - with respect to Test 3 - the resulting celerity is multiplied by 10, and the length of the cannel by 1000, the problem time is scaled 1000/10=100 times. Figure 5.3.c shows the comparison between the exact solution and the simulation, without friction (blue line) and including friction (green line, manning factor $n = 0.03$, see also Equation 1.3). The differences induced by the introduction of friction are very high in comparison to the differences induced by the simplicity of the method adopted to handle the dry bed state.

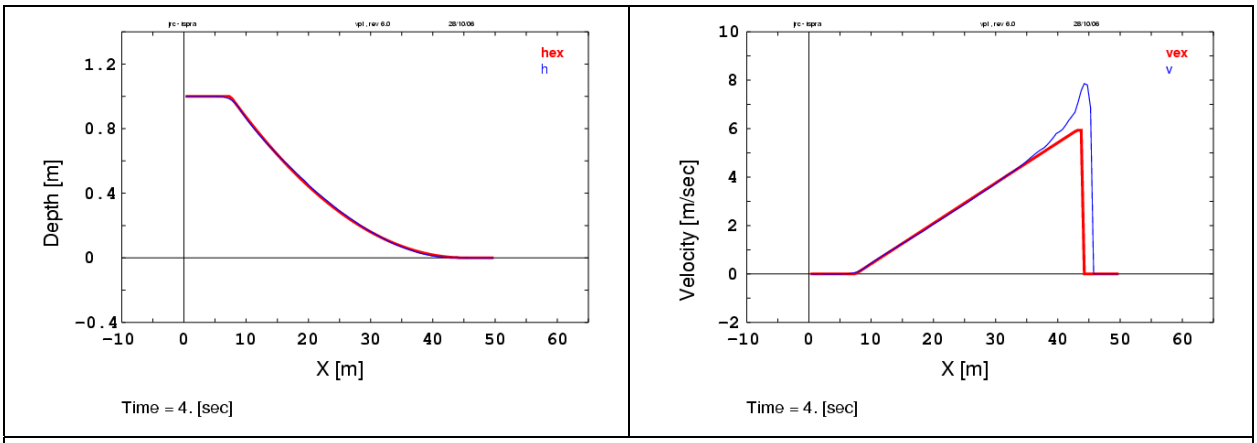


Figure 5.3.a – Test 3: Flux Vector Splitting Solver (blue line) and Exact solution (red line). Second order accuracy with MINMOD limiter function is applied (f=0).

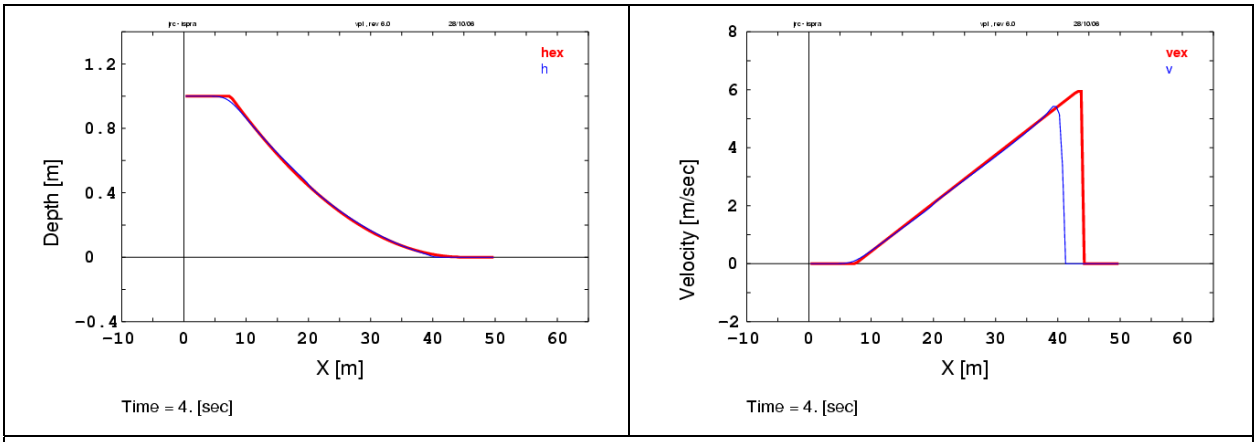


Figure 5.3.b – Test 3: Flux Vector Splitting Solver (blue line) and Exact solution (red line). Second order accuracy with MODIFIED MINMOD limiter function is applied (f=0.5).

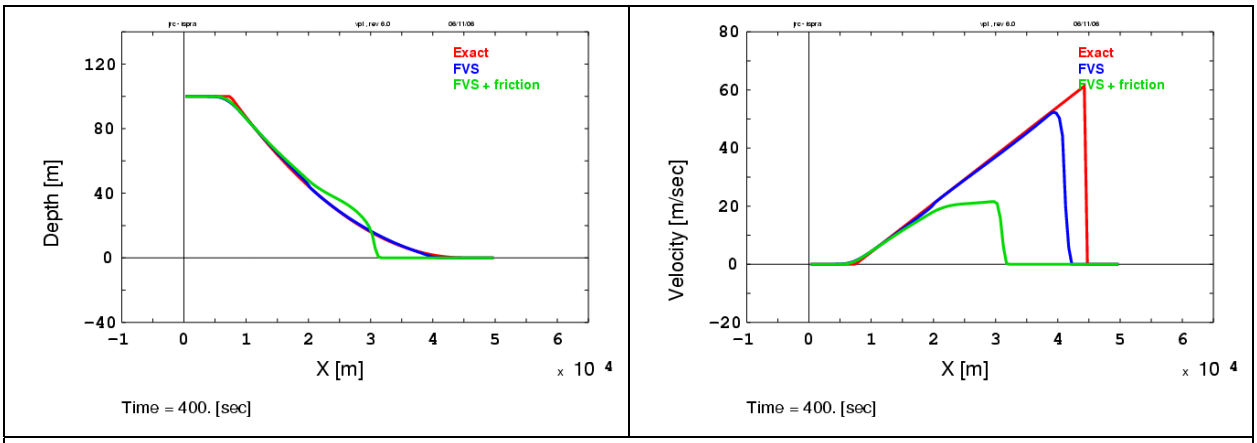


Figure 5.3.c – Test 3.1: Flux Vector Splitting Solver (blue line) and Exact solution (red line). FVS+Bed friction (green line). Second order accuracy with MODIFIED MINMOD limiter function is applied (f=0.5).

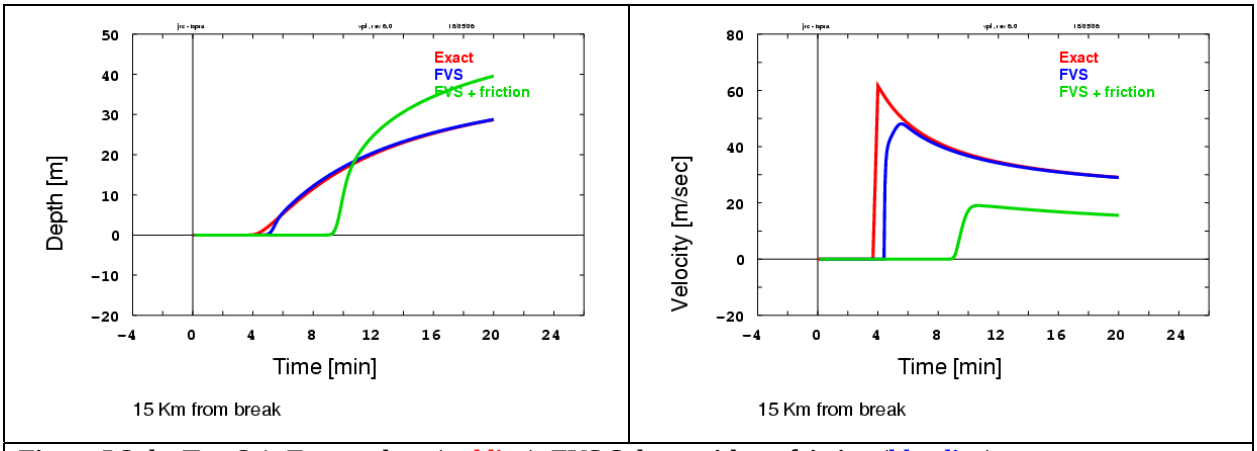


Figure 5.3.d – Test 3.1: Exact solver (red line), FVS Solver, without friction (blue line), FVS Solver + friction (green line)

Figures 5.3.d-e provides the water depth, velocity and flow rate ($Q = hv$) at 15 km after the break ($x = x_0 + 15 = 35km$) as function of time. The difference between the exact and FVS solver - without friction – of the wave arrival time is ~ 1 min. The water depth, velocity and flow rate are the same for the rest of the time. The error is significant only in the wetting process simulation. If the friction is included (as must be in a real case) the differences become more important: the arrival time is delayed more than 5 min, while the water depth is higher with respect to the inviscid case after ~ 2 min. The velocity and the flow rate are consistently lower for all the problem time. This leads to the conclusion that in defining the wetting process, the simple 'near dry bed' model adopted is enough; while care has to be taken in defining the bed friction coefficient.

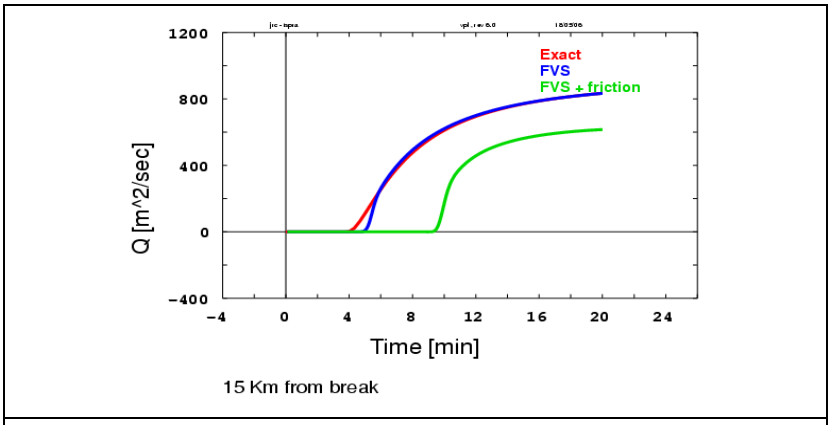


Figure 5.3.e – Test 3.1: exact solver (red line), FVS Solver, without friction (blue line), FVS Solver + friction (green line)

5.4 – Constant bed slope

This test case is chosen for the difficulties found by N.Adrianov [13] on solving numerically the problem: he found that, '*For certain initial conditions, the exact solution to the Riemann problem for shallow water equations in not unique. We test the performance of several numerical methods on such initial data and establish that the numerical solution can pick out different exact solutions*'.
In the figure below, the Exact Solution for an *inhomogeneous* equation have been derived from the 'Exact Solution' for *homogeneous* equations, following the work of Watson et al. [11] and re-proposed by F.Toro [8]. The procedure to obtain the Exact Solution for *inhomogeneous* equations, but with constant bed slope, is the following:

a) the Exact Solution is obtained as for the *homogeneous* equations (3.1)
b) the velocity profile and the x position are corrected following the change of variables suggested by Watson et al [11], obtaining the exact solution for the *inhomogeneous* equations (2.1)

$$
\begin{aligned}
t^{in\,\hom} &= t^{\hom} \\
x^{in\,\hom} &= x^{\hom} - \frac{1}{2}\frac{\partial z}{\partial t} g t^2 \\
h^{in\,\hom} &= h^{\hom} \\
v^{in\,\hom} &= v^{\hom} - \frac{\partial z}{\partial x} g t
\end{aligned}
\qquad (5.1)
$$

Note that replacing the variables of equations (5.1) into the *inhomogeneous* equations (2.1), the *homogeneous* equations (3.1) are obtained.

In Figure 5.4.a is a comparison of the numerical simulation with the derived exact solution. Note that the inlet and outlet velocities change during the transient: for the exact solution according to Equation (5.1), while for the simulation the velocities are updated according to the values in the neighbouring cells . The problems identified by Andronov [13] have not seen in the numerical simulations achieved by the FVS solver, nor in the calculation of the exact solutions.

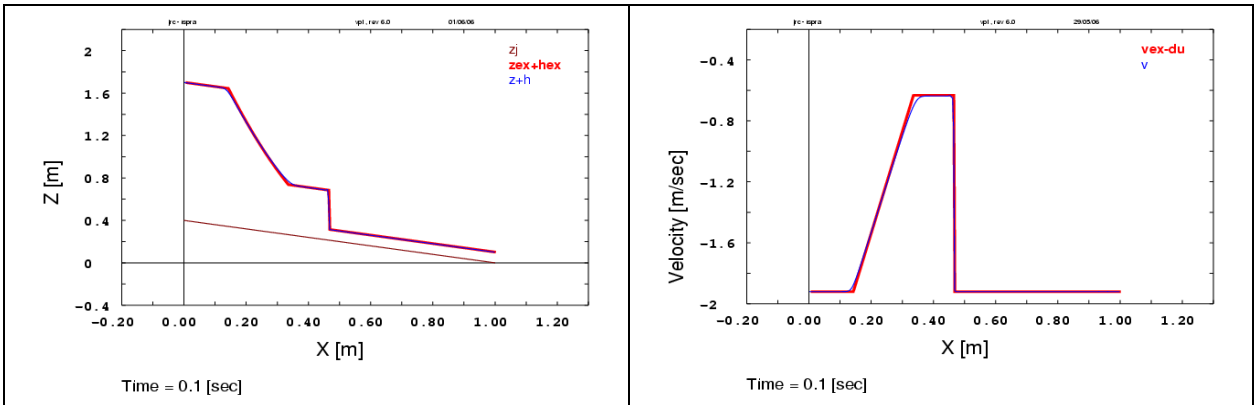


Figure 5.4.a – Test 4: Flux Vector Splitting Solver (blue line) and Exact solution (red line). Second order accuracy with MODIFIED MINMOD limiter function is applied (f=1).

5.5 – Bed step

Test 5 is a Dam Break problem with **bed step**, proposed by Jin and Wen [12] in order to test the suitability of their method to describe source terms concentrations, i.e., $\frac{\partial z}{\partial x} = \infty$. They solve the problems by subdividing the bed step into 5 sub steps and computing (by Newton's iteration) an interpolation function which satisfies momentum and energy continuity. The method applied here does not consider the bed step as a source term concentration, but a combination of free flowing stream and reflective boundary (Section 4.7). The space domain (20 m length) is subdivided into 100 cells: no sub steps are introduced at the bed step, i.e., the bottom slopes are set to zero in the entire domain in order to define the bed step.
The simulation results with 100 cells are compared with the 'exact' solution with 2000 cells, where the bottom step is modelled by a slope $\partial z / \partial x = 5$. In this case sources due to the bottom slope are modelled.

Figure 5.5.a shows the comparisons for the water depth, velocity, energy and momentum. The results are considerably good, given the simplicity of the adopted scheme to handle the bed steps. Note that the momentum is conserved at bed step while the energy due to the step is lost. No peaks are seen as in the results of Jin and Wen [12].

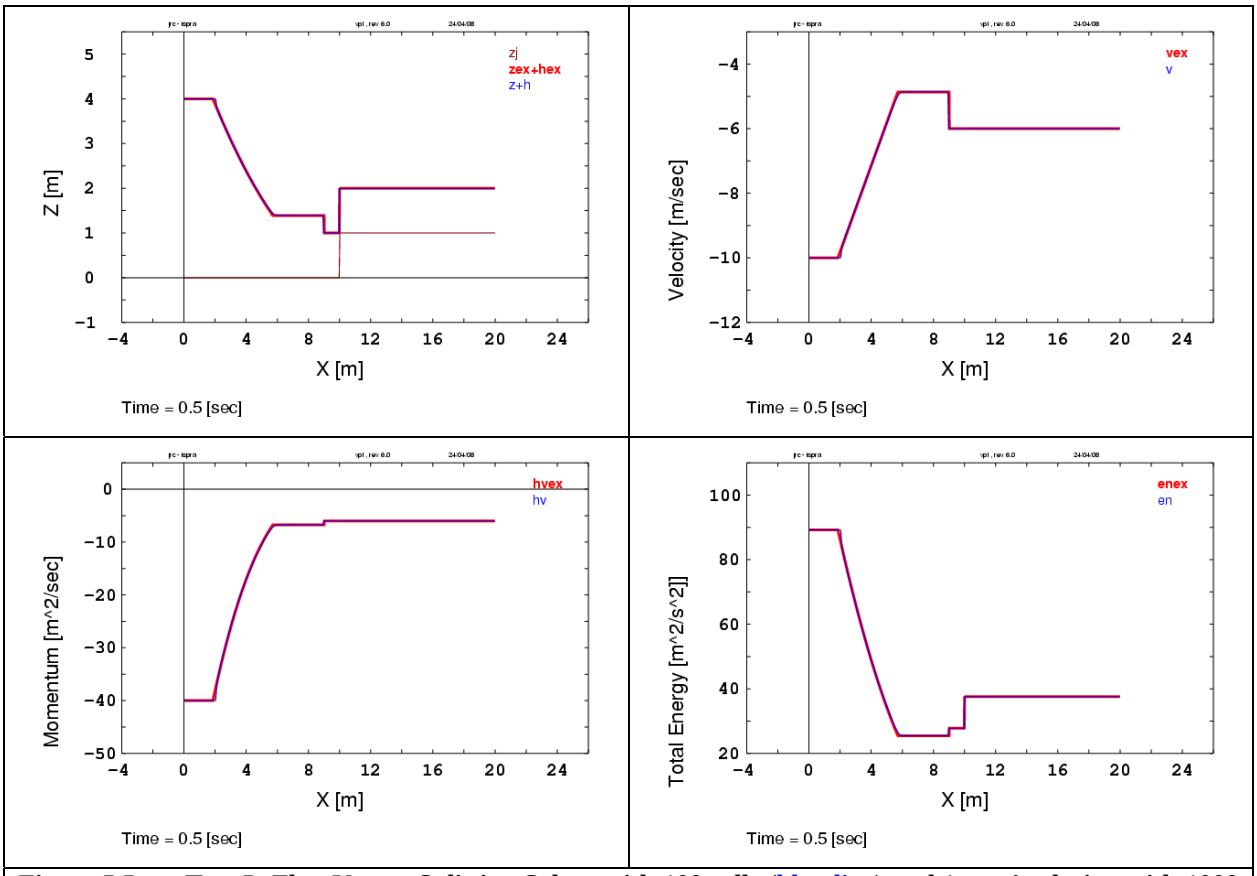
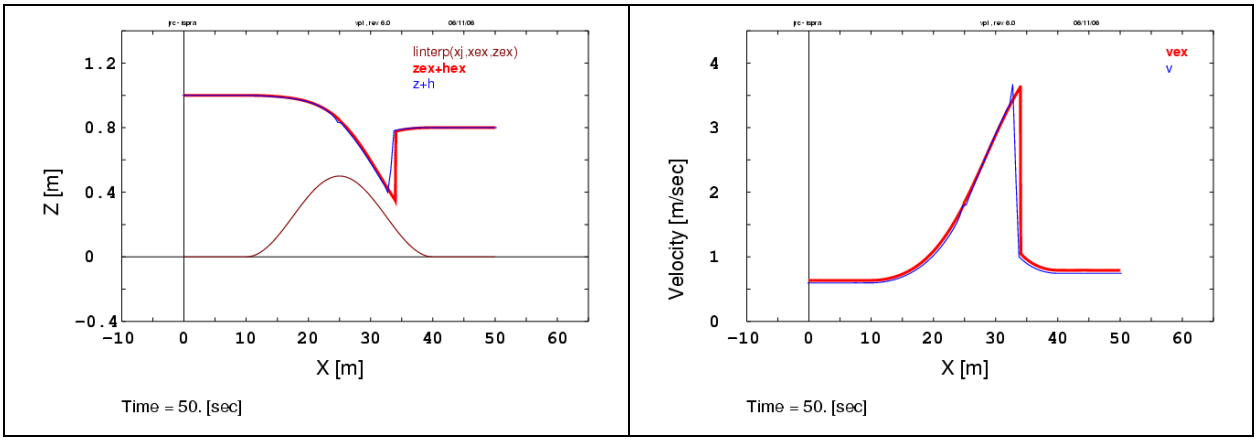


Figure 5.5.a – Test 5: Flux Vector Splitting Solver with 100 cells (blue line) and 'exact' solution with 1000 cells (red line). Second order accuracy with MODIFIED MINMOD limiter function is applied (f=1).

5.6 – Steady state problem

Test 6 is a steady state problem forced by critical flow conditions. Making a similarity with a nozzle problem in gas dynamic, the critical flow is established at maximum bed elevation (minimum throat for a nozzle). The maximum bed elevation is $z_0 = 0.5m$ at $x_0 = 25m$; the change in the bed elevation is performed by a sinus function, for $10m < x < 40m$. As for a nozzle in gas dynamic, the flow rate at the steady state is defined by the inlet water depth (pressure for the nozzle) and by the critical flow at the maximum bed elevation and not by the water depth (pressure for the nozzle) at the outlet. The steady state is reached after about 150 sec. As for Test case 5, the simulation results with 100 cells are compared with the 'exact' solution obtained by the same method with 1000 cells.

In Figure 5.6.a are the comparisons for the water depth, velocity, Froude no. and energy. The Froude no. is 1 exactly at x_0 and a bore is generated at $x_{bore} \cong 34m$ in order to establish a subsonic flow at the outlet, paid by a loss of energy. Note that for $x < x_0$ all the quantities are the same for the coarse and fine simulation, while small differences are seen for $x_0 < x < x_{bore}$, that disappear again for $x > x_{bore}$.



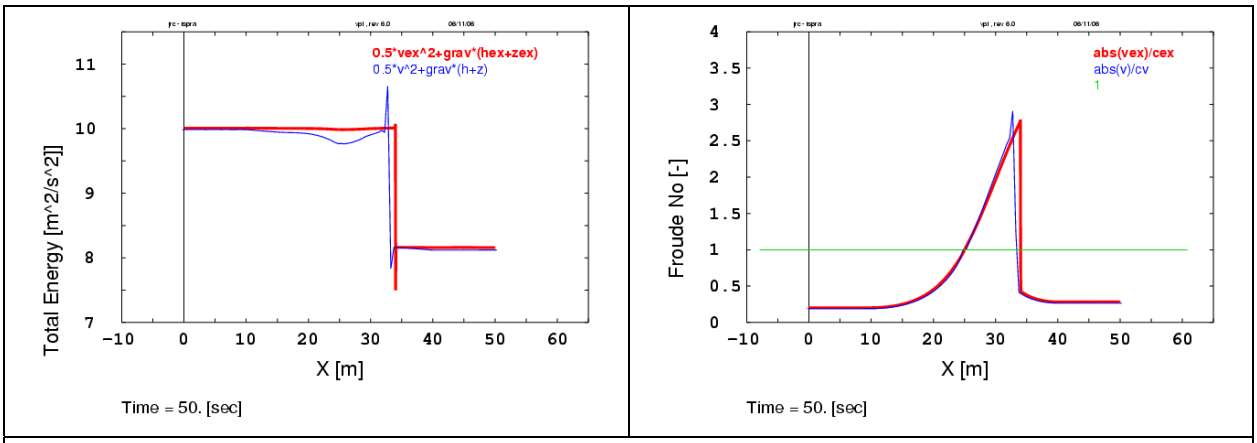


Figure 5.6.a – Test 6: Flux Vector Splitting Solver with 100 cells (blue line) and 'exact' solution with 1000 cells (red line). Second order accuracy with MODIFIED MINMOD limiter function is applied (f=1).

In Figure 5.6.b is the comparison for Test case 6.1, where the difference with Test 6 is only in the outlet water depth (0.5 m instead of 0.8m). As expected, the flow conditions for $x < x_0$ are the same as for the previous test because the inlet water depth and the maximum bed elevation are the same, while for the energy, loss occurs at the exit because the Froude No. established at the outlet is higher than 1, so the flow can accelerate like in a supersonic expansion in gas dynamic.

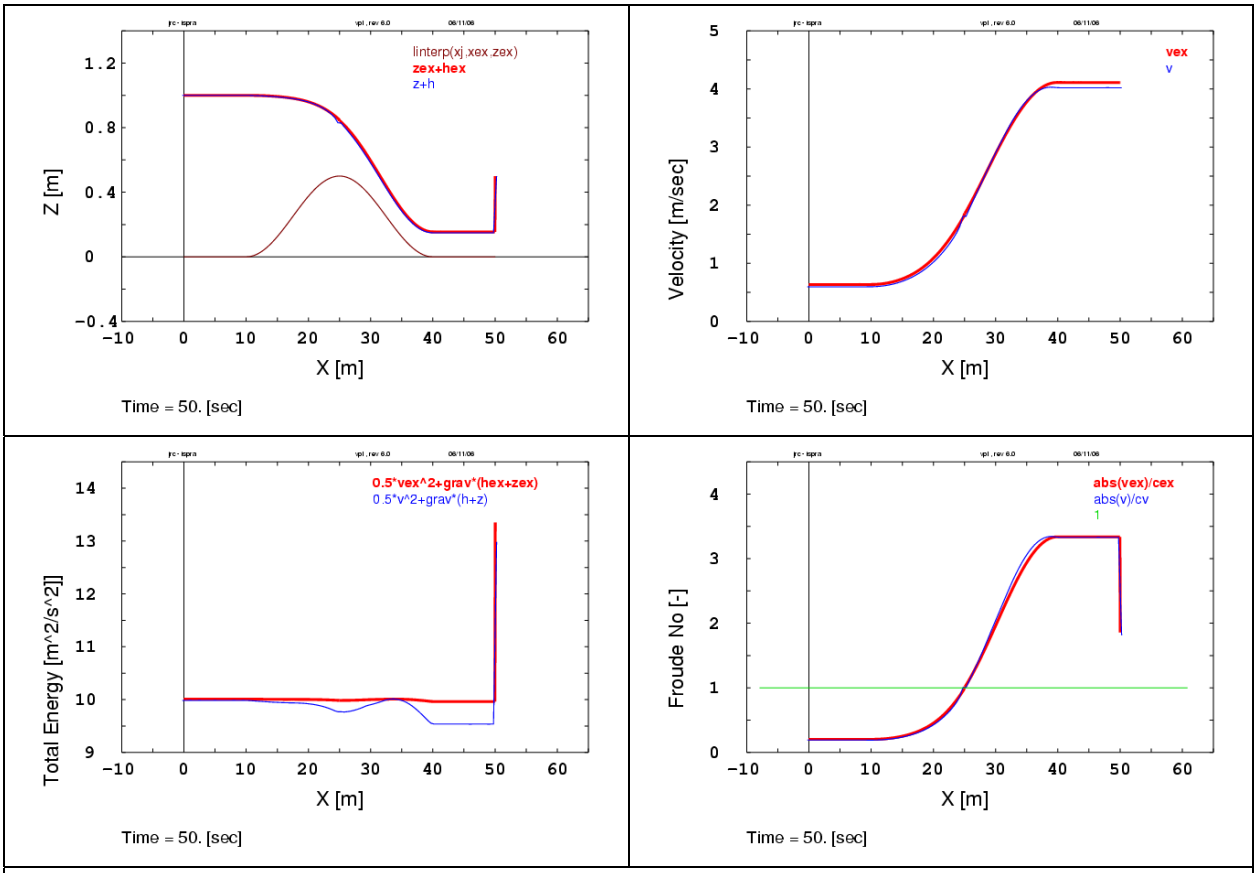


Figure 5.6.b – Test 6.1: Flux Vector Splitting Solver with 100 cells (blue line) and 'exact' solution with 1000 cells (red line). Second order accuracy with MODIFIED MINMOD limiter function is applied (f=1).

5.7 – Shoreline tracking

This is a simple shore line tracking problem, proposed as a benchmark exercise in "The third international workshop on long-wave runup models" [20]. The initial-value-problem technique introduced by Carrier, Wu and Yeh [21] is used to produce the benchmark data. The beach slope is fixed at 1/10 and the initial free surface elevation is provided in a text file, from 0 to 50 km, with 50 m steps. The initial velocity is assumed zero. The assignments of the modellers were to compute the temporal variations of the shoreline location and shoreline velocity. The snapshots of the free surface and velocity profiles at *t = 160 sec, 175 sec, and 220 sec*. were also requested. Such instants correspond to the time when the drying process ends (*t = 160 sec)* , the time during the rewetting phase when the shoreline moves with higher velocity (*t = 175 sec)* and finally the time when the wetting phase ends, i.e., the shoreline reaches the maximum elevation (*t = 220 sec)*. The exact solutions for such snapshots are provided with a space resolution of ~6 m for *t = 160 sec, 175 sec* and ~9 m for *t = 220 sec.*

Different simulations have been performed in order to validate the model with respect to the space resolution. The cell size is constants, i.e., no refinements have been performed near the shoreline. The simulations presented in the Fig. 5.7.a-b are obtained for *dx=200 m* (250 cells), *dx=50 m* (1000 cells) and *dx=10 m* (5000 cells). Fig. 5.7.a shows the comparisons of the shoreline position and velocity: we note that increasing the no. of cells the simulation approaches to the exact solution. It is of interest to note that in case of rough space resolution (*dx=200 m*) the position of the shoreline is captured with an error lower then the cell size.

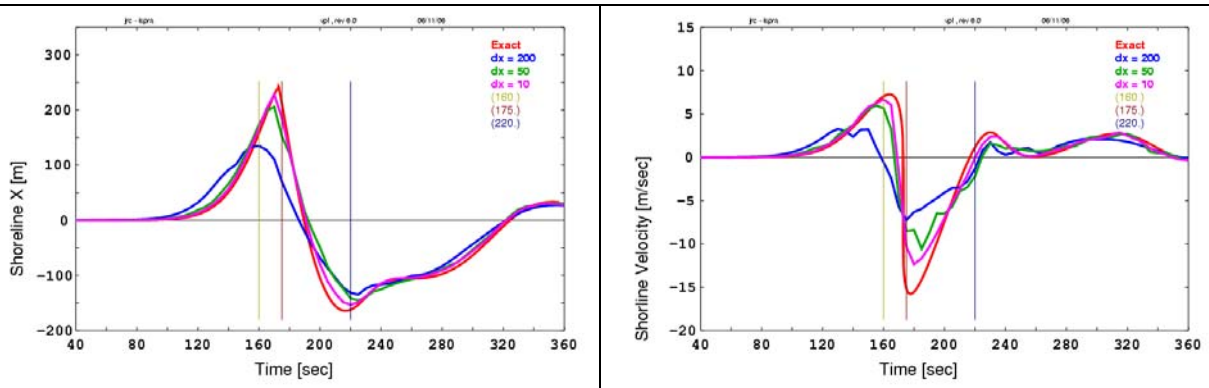


Figure 5.7.a – Test 7: Flux Vector Splitting Solver (blue, green, pink lines) and 'exact' solution (red line). Second order accuracy with MODIFIED MINMOD limiter function is applied (f=1).

Fig. 5.7.a shows comparison between the free surface elevations and velocity at different time. The free surface is well simulated for *dx=10 m*, while for *dx=200 m* the accuracy is not bad. However, despite the rough space discretisation when comparing with the scale of the exact free surface profile, the results are reliable and without unphysical behaviours, considering also that most of the numerical methods are bound to experience serious difficulties in resolving the flow in such conditions. The simulated velocity profile diverges from the exact solution in the proximity of the dry bed state. The major problems shown by any space resolutions are seen for the simulation of the rewetting process *(t=175 sec)*. Such deficiencies are recovered by a higher space resolution. The final stage of the wetting process *(t=220 sec)* is well captured by all the simulations shown in this paper.

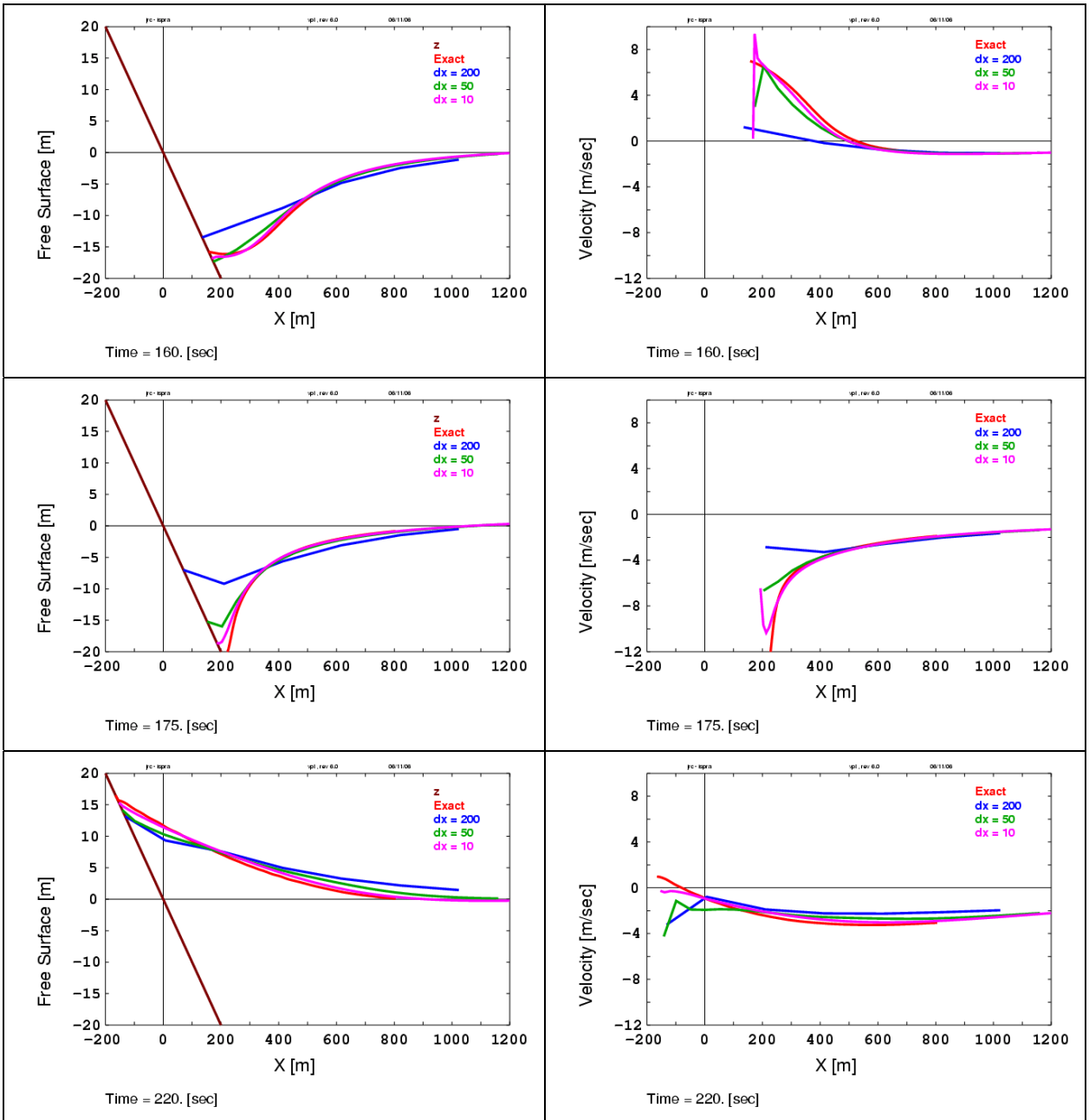


Figure 5.7.a – Test 7: Flux Vector Splitting Solver (blue, green, pink lines) and 'exact' solution (red line). Second order accuracy with MODIFIED MINMOD limiter function is applied (f=0.5).

Summary and Concluding Remarks

The objective of this paper has been to describe the method used to model the shallow water flow in changing bathymetry and topography. The hyperbolicity of the shallow water equations allows the algebraic evaluation of the eigenvalues and egenvectors. For the integration of the resulting governing equations, a finite volume, second-order Flux Vector Splitting technique is used, providing a high resolution of local flow conditions, such as dry as well wet bed, moving shoreline, critical flow, bottom steps.

The proposed model has been validated in respect to several numerical test cases for which an exact solution exist, proving the robustness and the reliability of the numerical schemes adopted. In particular we have shown the following:

- by means of the flux vector splitting technique, the approximate solution of the Riemann problem converges to the near exact solution without expensive iterations.
- the finite volume scheme has proved to be very efficient to handle "singularities" like critical flow, bed step and shoreline, conserving mass and momentum, without unphysical numerical sources that arise in most of the finite difference models.
- the second order scheme together with an appropriate limiter function allows a high resolution of the simulations, limiting unphysical numerical noise diffusion as well as numerical viscosity and allowing to converge to a nearly exact solution also in case of bore, critical flow and shoreline. However, small differences in respect to the exact solution exist for the drying processes, while the wetting process is well simulated.
- The rather simple models used to simulate the bed step and the shoreline tracking have been proved efficient and robust, allowing simulating inundation process also in case of coarse grid size.

The validation with 1 D test case has allowed focusing directly into the physics of the problems. The model has been implemented also in a 2 D system with real topography and bathymetry: the 2D model description together with test cases on real Dam break problems, tsunami propagation and long waver run-up problems will be presented in a future paper.

References

1. F. Ruel, Numerical Simulation of Reacting Gas Flows. Basic models and Algorithms. Presentation of a Preliminary Code Version – JRC Technical Note No.I.91.109 (1991)
2. H. Städtke, G. Franchello and B. Worth, Numerical Simulation of Multi-Dimensional Two-Phase Flow Using Second Order Flux Splitting Techniques, 7th Intern. Meeting on Nuclear Thermal Hydraulics, NURETH-7, Saratoga Springs, NY USA, September 10-15, 1995
3. H. Städtke, G.Franchello, B.Worth, Towards High Resolution Numerical Simulation of Transient Two-Phase Flow, Third International Conference on Multi-Phase Flow, ICMF'98, Lyon, June 8-12, 1998
4. H. Städtke, G. Franchello, B. Worth, 2003. Assessment of the JRC Advanced Two-Phase Flow Module using ASTAR Benchmark Test Cases, International Workshop on "Advanced Numerical Methods for Multidimensional Simulation of Two-Phase Flow, GRS-Garching, 15-16 September, 2003.
5. H. Städtke, Gasdynamic aspects of Two-Phase Flow. Hyperbolicity, Wave Propagation Phenomena, and Related Numerical Methods. WILEY-VCH Verlag GmbH & Co. KGaA, 2006
6. V.V. Titov and F.I. Gonzalez, **Implementation and Testing of the Method of Splitting Tsunami (MOST) Model**,. NOAA Technical Memorandum ERL PMEL-112, November 1997.
7. E. F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics. Springer-Verlag, Berlin/Heidelberg/New York, 1997, ISBN 3-540-65966-8. Second edition 1999.
8. E. F. Toro, Shock-Capturing Methods for Free-Surface Shallow Flows. John Wiley and Sons, Chichester, 2001, ISBN 0-471-98766-2.
9. E. F. Toro (Editor), Godunov Methods: Theory and Applications. Edited Review. Kluwer Academic/Plenum Publishers, 2001.
10. E. F. Toro, NUMERICA, A Library of Source Codes for Teaching, Research and Applications, Numerik Ltd., www.nuymerika.com, 1999
11. G.Watson, D.H Peregrine, and E.F. Toro. Numerical Solution of the Shallow Water equation on a Beach Using Weighted Average Flux Method. Computational Fluid Dynamics 1992, Vol1, pages 495-502. Elsevier, 1992
12. Shi Jin and Xin Wen, An efficient method for computing hyperbolic systems with geometrical source terms having concentrations, J. Comp. Math. 22, 230-249, 2004.
13. N. Andrianov, "Performance of numerical methods on the non-unique solution to the Riemann problem for the shallow water equations", Int. J. Num. Meth. Fluids 2005; 47:825-831.
14. R.J. LeVeque and D.L. George. High-Resolution Finite-Volume Methods for the Shallow Water Equations with Topography and Dry-States. Proceedings of the 3rd International Long-Wave Workshop, Catalina, June 2004.
15. Scott F. Bradford and Nikolaos D. Katopodes, Finite Volume Model for Nonlevel Basin Irrigation, J. Irrig. and Drain. Engrg., Volume 127, Issue 4, pp. 216-223 (July/August 2001)
16. Randall J. LeVeque, Finite Volume Methods for Hyperbolic Problems, Series: Cambridge Texts in Applied Mathematics (No. 31), University of Washington
17. Audusse E, Bouchut F, Bristeau MO, Klein R, Perthame B. A fast and stable well-balanced scheme with hydrostatic reconstruction for shallow water ows. SIAM J.Sci.Comp. 2004; 25(6):2050-2065.
18. Fabien Marche Theoretical and Numerical Study of Shallow Water Models. Applications to Nearshore Hydrodynamics Thesis Université de Bordeaux, http://www.math.u-bordeaux.fr/~marche/THESE_Marche.pdf
19. XinJian Chen, A Cartesian method for fitting the bathymetry and tracking the dynamic position of the shoreline in a three-dimensional, hydrodynamic model, Journal of Computational Physics, Volume 200, Issue 2, , 1 November 2004, Pages 749-768.
20. "The third international workshop on long-wave runup models", June 17-18 2004, Wrigley Marine Science Center, Catalina Island, California, http://www.cee.cornell.edu/longwave/
21. Carrier, Wu and Yeh , Journal of Fluid Mechanics, 475, 79-99, 2003
22. P. Brufau and P. García-Navarro, Unsteady free surface flow simulation over complex topography with a multidimensional upwind technique, Journal of Computational Physics, Volume 186, Issue 2, 10 April 2003,Pages 503-526
23. J. G. Zhou, D. M. Causon, C. G. Mingham and D. M. Ingram, The Surface Gradient Method for the Treatment of Source Terms in the Shallow-Water Equations, Journal of Computational Physics, Volume 168, Issue 1, 20 March 2001, Pages 1-25
24. Randall J. LeVeque, Balancing Source Terms and Flux Gradients in High-Resolution Godunov Methods: The Quasi-Steady Wave-Propagation Algorithm , Journal of Computational Physics, Volume 146, Issue 1, 10 October 1998, Pages 346-365

European Commission
JRC 35059
EUR 23307 EN
ISSN 1018-5593

EUR 23307 EN – Joint Research Centre – Institute for the Protection and Security of the Citizen
Title: Modelling shallow water flows by a High Resolution Riemann Solver
Author: G.Franchello
Luxembourg: Office for Official Publications of the European Communities
2008 – 28 pp. – 29.7 x 21 cm
EUR – Scientific and Technical Research series – ISSN 1018-5593

Abstract
Shock-capturing methods originally developed for compressible gas dynamics have been applied to many other non-linear hyperbolic systems of conservation equations, like reactive flows, two-phase flow, porous media flow and finally shallow water flows, with applications to rivers, estuaries, dam break and in particular tsunami propagation resulting from earthquake and landslide.

The basic ingredient of the proposed model for solving the shallow water equations is a finite volume Flux Vector Splitting Riemann Solver with a second order resolution scheme and implicit treatment of the source terms. The model includes entropy fix, bed step treatments, and finally shore line tracking, giving the ability to capture local discontinuities - like shock waves - and reducing numerical diffusion and unphysical viscosity effects which dominates in all finite difference methods. The numerical model has been validated in respect to different numerical test cases. Furthermore, comparisons with the exact solution of the Riemann problems are presented.

The mission of the JRC is to provide customer-driven scientific and technical support for the conception, development, implementation and monitoring of EU policies. As a service of the European Commission, the JRC functions as a reference centre of science and technology for the Union. Close to the policy-making process, it serves the common interest of the Member States, while being independent of special interests, whether private or national.